



PIZZACRAFT

It is a long-standing tradition on many programming competitions to provide the competitors with some delicious PIZZA. However, during last year's *Programming Intricacies and puzzling Algorithms* competition, the pizza delivered was, weirdly enough, cut into an odd number of slices. Teams, all ravenous, fought for the remaining slices. Since all competitors are skilled programmers, they decided to resolve the dispute by challenging each other in a strategy game - controlled by hand-crafted, state-of-the-art programs, obviously.

(Note: You **will** be given pizza regardless of how well you perform in this game).

Overview

This is a turn-based strategy game. Players mine gold, train units, construct buildings and attack each other. The goal is to conquer the enemy by destroying all their buildings.

In every match exactly two teams face each other. At the same time multiple matches are played simultaneously, you can expect to play two matches with all other teams at the same time.

Players can only perform actions on their own turn. There is a turn limit per match - if no player wins the game within the limit, the match ends anyway. All matches start at the same time, so even if you won all your matches, you may need to wait for a bit until the next round starts.

The game is played on a hex grid of a fixed, finite size. The center tile is at $(0, 0, 0)$. The map size is specified by its radius, that is, the map of size N consists of all tiles of a hex grid whose distance to the center is less or equal to N . The game uses **cubic coordinates** system (see Appendix B).

There are two core resources that are needed to build you army: Gold and Supplies.

Gold is gained by ordering your workers to work in a gold mine. The amount of gold available in mines is finite. You spend gold to train new units and construct new buildings.

Supplies are provided by houses and headquarters. Each unit consumes a fixed number of supplies. They are, however, reclaimed when the unit is destroyed. Your number of supplies effectively imposes a limit on your army size.

All units have a number of stamina points at their disposal during your turn. Moving units consumes 1 stamina point. Many actions, such as building or mining, require and consume all stamina points available to the unit (so the unit must not have moved in the same turn). At the beginning of your turn, stamina points of all your units are replenished.

There is a fog of war over game field. You can't tell what is your opponent doing, unless you get close enough to them - your units and buildings have a limited range of sight.

Unit types

There are three unit types available:

- 1 - Worker - can mine gold and initiate building construction
- 2 - Warrior - can attack adjacent units and buildings
- 3 - Archer - can attack adjacent or distant units and buildings

Building types

There are five building types available:

- 4 - Headquarters (HQ) - trains workers, provides supplies
- 5 - Barracks - can train warriors (and archers, if you have armory)
- 6 - House - provides supplies
- 7 - Outposts - has a long range of sight
- 8 - Armory - required to build archers, can develop upgrades

Movement

All units can move, buildings are stationary. No unit can enter tiles with water, gold mines, buildings, or tiles where enemy units stand.

You can stack up to 3 units on the same tile.

Units move only by one tile at a time. However, units with more than 1 stamina point can move multiple times in a single turn, extending their effective movement range depending on their available stamina.

Sight

All units and buildings have a specified range of sight. You know the location of water tiles, gold mines, your own units and buildings, but you can only see enemy units and buildings if they are within the range of sight of one of your units or buildings.

Mining gold

There are gold mines on the map, and workers can use them to get you gold. This action requires and consumes all stamina points available to the unit. The worker must stand on a tile adjacent to the mine. Additionally, you must have an HQ building within 2 tiles of the gold mine. There is a finite amount of gold in each mine. You can query how much gold is left in the mine if it is within the range of sight of one of your units or buildings.

Training units

Headquarters can train new workers, barracks can train new warriors and archers. Buildings train units on a selected adjacent tile - which must satisfy the requirements for unit movement (no water, buildings, gold mines, no enemy units, at most 2 friendly units). To begin training you need to pay gold in amount depending on unit type, but you must also have enough supply units available. A building can only train one unit at a time.

When you order a building to train a unit, it appears instantaneously. However, it is unable to move and act until it completes its training, which usually takes a few turns. The new unit starts with full HP, as soon as it appears on the board.

There is no limit for the number of units of any type.

Building construction

Workers can initiate building construction from any adjacent tile, though they don't need to be present for the building to continue construction. The 'build' action requires and consumes all stamina points available to the worker. To start building construction you need to spend gold, the amount depending on building type. While the building is constructed, it cannot perform any actions.

The number of buildings you can build is not limited, regardless of building type.

Attack

Warriors and archers can attack enemy units and buildings. An attack requires and consumes all stamina points available to the unit. All units and buildings have armor, which decreases the attack power.

The damage dealt to the attacked unit is computed by subtracting its armor value from the attacker's attack value. However, all attacks deal at least one damage. For example, if the attacker has 6 attack points and the attacked unit has 3 armor points, it will receive 3 damage. If the attacker has 5 attack points and the attacked unit has 10 armor points, it will receive 1 damage.

The received damage is subtracted from attacked unit's or building's health points. If its health points reach zero or less, the unit/building is destroyed.

Health regeneration

If a unit has less than maximum health points, it may slowly regain health. For this to happen, it must have been uninvolved in combat (both attacking and being attacked) for at least four turns (of any player). The unit will then regenerate two health points each turn (both your and enemy turns) until it participates in combat or reaches maximum HP.

Upgrades

You can use your armory to develop upgrades, which increase attack or armor parameters of your units. You spend gold to develop upgrades. They are applied immediately to both new and pre-existing units. However, the armory that developed the upgrade will need to cool down for a few turns, and will be unable to develop next upgrades. You can build archers in barracks during armory cooldown.

There are two types of upgrades available:

- Type 0 - increases attack of your warriors and archers by 1
- Type 1 - increases armor of all your units by 1

You can develop the same type of upgrade multiple times, up to the maximum of 10 times (per upgrade type) - their effects stack.

The gold cost of developing new upgrades is computed using the formula: $50 * 1.2^n$, rounded down, where n is the current level of this upgrade that you already have developed. This means that the first upgrade costs 50 gold. Costs of both upgrade types are computed independently. The armory cooldown time can be computed using a similar formula: $10 * 1.2^n$.

Starting conditions

You are guaranteed to have at least one worker available when the match starts. You will also have at least one headquarter within mining distance (2) from a gold mine. The game board (map) will always be symmetrical.

Scoring

In each match there is 1000 match points to be divided between both players.

If one player wins by destroying all enemy building, they receive all 1000 points, and the defeated player receives 0.

If the turn limit is reached without a decisive win, the match points are divided between players proportionally to the points gained by destroying enemy buildings, according to the table below.

Example: Player A destroyed enemy house and barracks (3 + 10). Player B destroyed 4 enemy houses (4 * 3). Player A will receive 520 ($1000 * 13 / (13 + 12)$) points, player B will receive 480 ($1000 * 12 / (13 + 12)$).

Detailed parameters

	Worker	Warrior	Archer		Headquarters	Barracks	House	Outpost	Armory
Price to train	15	35	25	Price to build	80	50	20	30	60
Training time	4	6	6	Time to build	12	8	6	6	10
Range of sight	2	3	3	Range of sight	3	2	1	5	2
Stamina	2	2	2	Maximum HP	50	40	25	18	40
Maximum HP	20	32	27	Armor	6	4	2	3	4
HP regen. per turn	2	2	2	Supplies provided	15	-	9	-	-
Attack	-	10	7	Points for destroying	30	10	3	2	15
Attack range	-	1	3						
Armor	0	2	1						
Supplies used	2	3	3						

Commands

In the examples, > represents commands send by the player to the game server, and < represents commands send by the game server to the player.

Unless explicitly stated otherwise, you can only use commands on your turn.

All commands send by players immediately receive an OK response, if the command was correct and accepted. Otherwise the server responds with ERR num text, where num is the error code, and text is a human-readable explanation of the error.

WAIT

Responds with OK, then waits until next turn begins, and responds with OK NEXT_TURN. When a new match begins, and the new turn is the first turn of a new round, the second response is OK NEXT_ROUND instead. You can use this command regardless of whether it is your turn.

```
> WAIT
< OK
(after some time...)
< OK NEXT_TURN
```

GET_MY_MATCHES

Returns the number of matches you are currently participating in, and then a list of IDs of these matches. You can use this command regardless of whether it is your turn.

```
> GET_MY_MATCHES
< OK
< 4
< 1442
< 911
< 305
< 1290
```

IS_MY_TURN match_id

Responds with YES when it is your turn to make a move in this match, and NO if it is your opponent's turn. You can use this command regardless of whether it is your turn.

```
> IS_MY_TURN 1442
< OK
< YES
```

GET_TURNS_LEFT match_id

Returns the number of turns left till the turn limit in the specified match. This is the total number of turns, including your and your opponent turns. This means that if this command responds with 50, then you will have 25 turns at your disposal in this match.

You can use this command regardless of whether it is your turn or not.

```
> GET_TURNS_LEFT 12
< OK
< 50
```

GET_MAP

Responds with the description of the current map. All simultaneous matches use the same map, but it may change when a new round starts.

First number of the response is the map size (hex board radius).

Then follows the number of water tiles, and a list of their coordinates.

Next is the number of gold mines on the map.

Finally, there is a list of parameters per each gold mine - consisting of their ID, X, Y, Z coordinates, and the amount of gold that was available in this mine at the beginning of the match.

You can use this command regardless of whether it is your turn or not.

```
> GET_MAP
< OK
< 4
< 6
< 0 -4 4
< 0 4 -4
< 4 0 -4
< -4 0 4
< 4 -4 0
< -4 4 0
< 1
< 0 -2 1 1 10000
```

GET_GOLD match_id

Responds with your current amount of gold. You can use this command regardless of whether it is your turn or not.

```
> GET_GOLD 1442
< OK
< 10000
```

GET_SCORE match_id

Responds with the current score you would receive if the match was to end right now. This is just an approximation rounded to nearest integer. Your opponent will receive the remaining points of the total 1000 to be distributed in this match.

If you destroy all enemy buildings, the match ends and you receive all 1000 points.

Your final score may vary slightly, if you or your enemy destroy some buildings after you called GET_SCORE for the last time before match end.

You can use this command regardless of whether it is your turn or not.

```
> GET_SCORE 700
< OK
< 520
```

GET_MINES match_id

Responds with N - the number of mines that are in your range of sight.

Then, N lines follow. Each line consists of two numbers, first is the ID of the gold mine, and second is the amount of gold left in the mine.

To get a full list of all mines on the map, use GET_MAP instead.

```
> GET_MINES 411
< OK
< 3
< 0 10000
< 1 9920
< 2 0
```

GET_MY_UNITS match_id

Responds with N - the number of your units. Then N lines follow, each uses the format:
ID X Y Z TYPE TRAIN_TIME HP STAMINA

- ID is the ID of the unit.
- X Y Z are the coordinates of the tile this unit stands on.

- TYPE is the type of this unit (1 - worker, 2 - warrior, 3 - archer).
- TRAIN_TIME is the number of turns left until the unit completes the training. A unit cannot move or act while it is training. If a unit is no longer training, this number is 0.
- HP are the current health points of this unit.
- STAMINA is the number of stamina points this unit can still spend in this turn.

```
> GET_MY_UNITS 1442
< OK
< 2
< 1 1 -1 0 1 0 20 2
< 1 2 -2 0 2 4 50 2
```

GET_ENEMY_UNITS match_id

Responds with N - the number of enemy units.

Then N lines follow, each uses the format:

```
ID X Y Z TYPE TRAIN_TIME HP
```

- ID is the ID of the unit.
- X Y Z are the coordinates of the tile this unit stands on.
- TYPE is the type of this unit (1 - worker, 2 - warrior, 3 - archer).
- TRAIN_TIME is the number of turns left until the unit completes the training. If the unit is no longer in training, this number is 0.
- HP are the current health points of this unit.

```
> GET_ENEMY_UNITS 511
< OK
< 2
< 1 1 -1 0 1 0 20
< 1 2 -2 0 2 4 50
```

GET_MY_BUILDINGS match_id

Responds with N - the number of your buildings.

Then N lines follow, each uses the format:

```
ID X Y Z TYPE HP BUSY_TIME CONSTRUCTION_TIME
```

- ID is the ID of the building.
- X Y Z are the coordinates of the tile this building stands on.
- TYPE is the type of this building (4 - headquarters, 5 - barracks, 6 - house, 7 - outpost, 8 - armory)
- HP are the current health points of this building.
- BUSY_TIME is the number of turns this building will be busy for. This includes item training time (for barracks and headquarters) and upgrade cooldown time (for armories). If the building is not busy, this will be 0.
- CONSTRUCTION_TIME is the number of turns until this building completes construction. If construction has already finished, this will be 0.

```
> GET_MY_BUILDINGS 511
< OK
< 2
< 5 -2 2 0 5 40 0 8
< 4 -1 0 1 4 50 0 0
```

GET_ENEMY_BUILDINGS match_id

Responds with N - the number of enemy buildings that are currently within your range of sight. Then N lines follow, each uses the same format as lines in GET_MY_BUILDINGS.

```
> GET_ENEMY_BUILDINGS 1442
< OK
< 2
< 5 -2 2 0 5 40 0 8
< 4 -1 0 1 4 50 0 0
```

GET_UPGRADES match_id

This command responds with four numbers: Y0 Y1 E0 E1.

Y0 and Y1 are upgrade levels (type 0 and 1 respectively) you have developed.

E0 and E1 are upgrade levels (type 0 and 1 respectively) your enemy has developed. In order to know enemy upgrade levels you must have at least one enemy unit (of any type) in your range of sight. Otherwise E0 and E1 will be -1.

```
> GET_UPGRADES 1442
< OK
< 5 4 -1 -1
```

MOVE match_id unit_id x y z

Orders unit `unit_id` to move to position `x y z`. Unit must have at least one stamina point left to move. Moving consumes 1 stamina point.

Units can only move to an adjacent tile (but if they have enough stamina you can repeat the MOVE command multiple times in a turn).

Units cannot be moved onto tiles with water, buildings, enemy units or mines.

There can be at most 3 units on the same tile.

```
> MOVE 1442 5 0 -1 1
< OK
```

MINE match_id unit_id mine_id

Orders your unit `unit_id` to mine gold from gold mine `mine_id`. Only workers can mine gold. You must have a headquarters building within 2 tiles of the mine to be able to get gold from that mine.

The worker must stand on a tile adjacent to the mine and must have all of its stamina points available (it couldn't move earlier in this turn). This action consumes all stamina points of the unit.

This action increases your available gold by 5.

```
> MINE 1442 5 1
< OK
```

TRAIN match_id building_id unit_type x y z

Orders your building `building_id` to create a new unit of type `unit_type` on tile `x y z`. This requires you to spend gold, the cost is specified in unit parameters table. The target tile must be adjacent to the building and must satisfy the requirements for unit movement.

The new unit immediately has full health points. It is unable to move or act for several turns after creation, the exact number is specified as 'training time' in unit parameters table. During this period the building cannot train any other units. Note that the building stays inactive (cannot perform any actions) even if the trained unit gets destroyed before it completes its training, and, similarly, if the building is destroyed, the unit will continue training.

```
> TRAIN 1442 20 1 3 0 -3
< OK
```

BUILD match_id unit_id building_type x y z

Orders your unit `unit_id` to initiate building `building_type` construction on tile `x y z`. Only workers can construct buildings. This action spends gold required to build the building of this type. The target tile must be adjacent to the unit, and it must be completely empty (no water, mine, buildings nor units).

The worker must have all stamina points available (it can't have moved earlier in this turn). This action consumes all stamina points of the worker.

The building takes some time to construct during which it is inactive. The exact times are specified in the unit parameters table. The worker does not need to stay around while the building completes construction.

```
> BUILD 1442 5 6 4 -4 0
< OK
```

ATTACK match_id unit_id target_id

Orders your unit `unit_id` to attack target `target_id`. Only warriors and archers can attack. The target must be an enemy unit or enemy building. The target must be within attack range from the attacking unit.

The attacker must have all stamina points available (it can't have moved earlier in this turn). This action consumes all stamina points of the attacking unit.

```
> ATTACK 1442 9 14
< OK
```

UPGRADE match_id building_id upgrade_type

Orders building `building_id` to develop upgrade type `upgrade_type`. This action costs gold. Only armories can develop upgrades. The upgrade is applied immediately. The armory becomes inactive while it cools down.

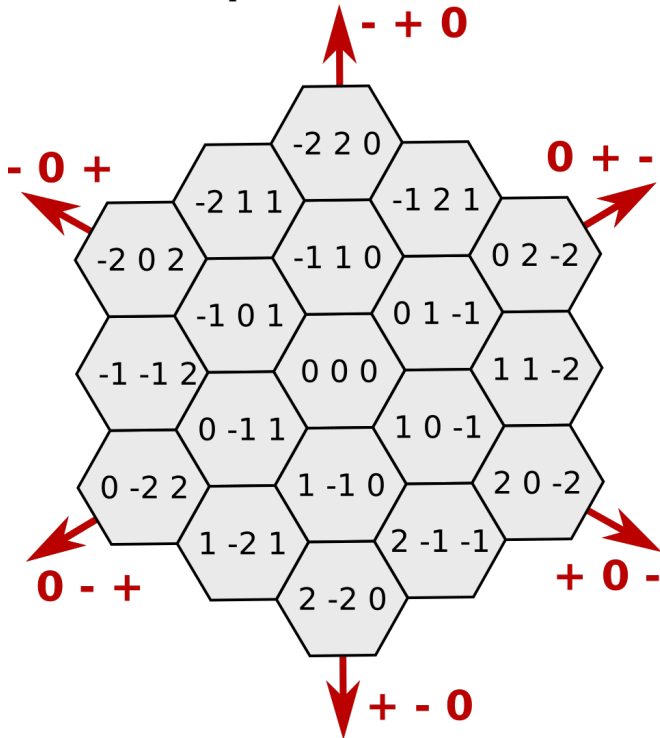
```
> UPGRADE 1442 21 1
< OK
```

Appendix A - Error codes list

- 102 - Commands limit reached
- 103 - Unknown command
- 104 - Invalid arguments
- 401 - Not your match
- 402 - Not your turn
- 501 - Not your unit
- 502 - Only workers can build
- 503 - Only archers/warriors can fight
- 504 - Not enough stamina
- 505 - Invalid unit type
- 506 - Position is out of map or coordinates are invalid
- 507 - Invalid upgrade type
- 508 - Enemy unit on given position
- 509 - Too many units on the same tile
- 510 - You've reached max. level of this upgrade
- 601 - Not your building
- 602 - Only headquarters can train workers
- 603 - Only barracks can train archers/warriors
- 604 - Only armory can develop upgrades
- 605 - Building is busy
- 606 - You need armory to train archers
- 701 - Building/unit is still training
- 702 - Target is not enemy unit/building or is out of sight
- 802 - Worker can build only in range exactly 1
- 803 - Invalid building type
- 804 - Unit/Building is out of range
- 805 - The tile has an obstacle
- 806 - Unit can move only in range of 1
- 807 - Mine with given ID does not exist
- 808 - Mine too far from any of your headquarters
- 809 - Spawning point is too far from building
- 901 - Not enough gold
- 902 - Not enough supplies

Appendix B - cubic hexagonal coordinates

With cubic coordinates, each tile is addressed by three numbers (X, Y, Z) . This is what a map of size 2 looks like:



Moving from a tile to an adjacent one requires adding 1 to one coordinate and subtracting 1 from another. You can do that in 6 ways, which corresponds to 6 directions of movement. This implies that all valid coordinates have the sum $X + Y + Z$ equal to zero.

Therefore, the neighbors of tile (XYZ) are: $(X+1, Y, Z-1)$, $(X-1, Y, Z+1)$, $(X+1, Y-1, Z)$, $(X-1, Y+1, Z)$, $(X, Y+1, Z-1)$, $(X, Y-1, Z+1)$.

The distance between tiles (A_x, A_y, A_z) and (B_x, B_y, B_z) can be computed using this formula:

$$\max(|A_x - B_x|, |A_y - B_y|, |A_z - B_z|)$$