

Principle:

Components:-

- A simulator program (Server) which has been written by the organizing team of Edge.
- C/C++/Java (Client) code submitted by you within a given template.

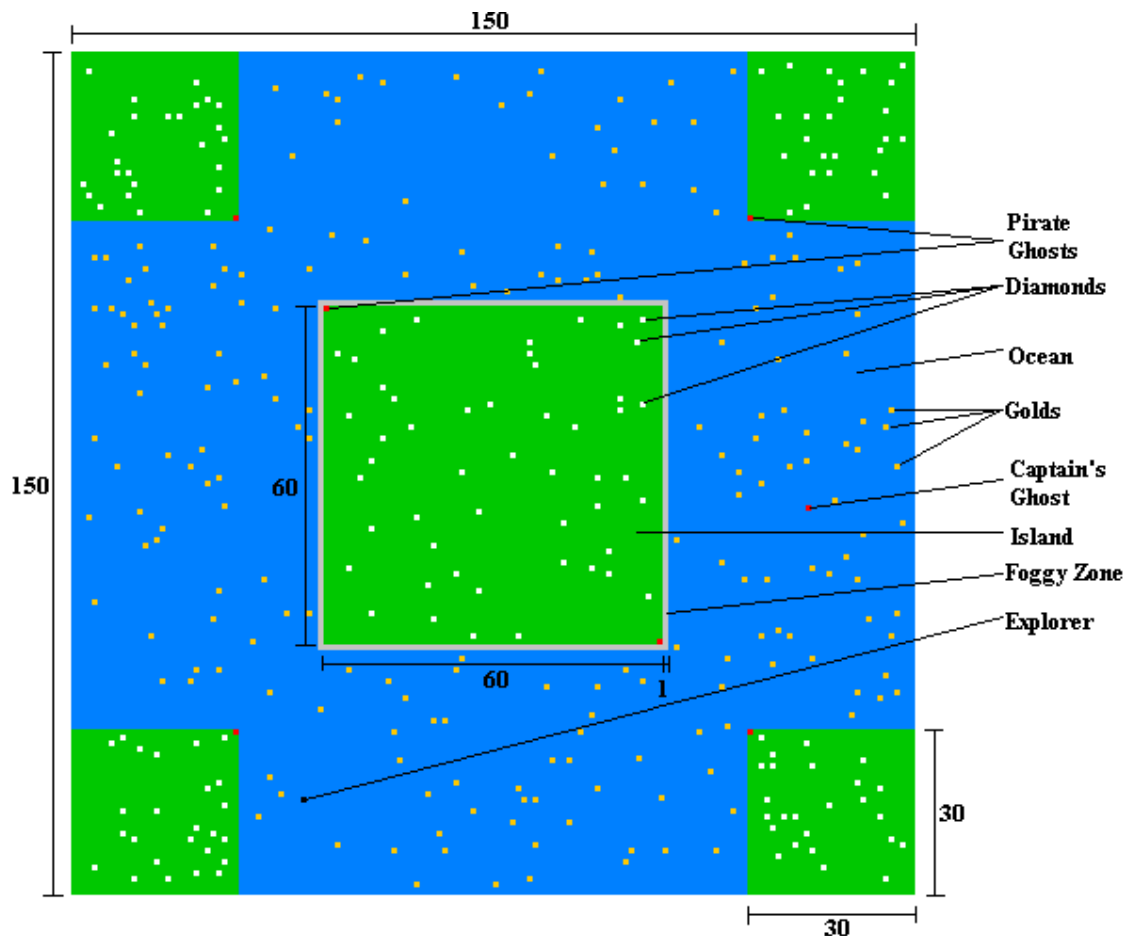
Once the simulator starts, you will be provided with the visual representation of a map. Then:-

- When you run your Client code, it will connect to the simulator (Server), and then the simulator will start interacting with your code by sending environment information and receiving request. The 'Explorer' will be controlled by your code.
- A 'cycle' of time period 1 second is started. The simulator (Server) waits for request from Client during this time. At the end of 1 second, it will process the request, check validity of the request, serve the request and return the updated environment information to the client.
- This entire process is clubbed as a 'cycle'. The program will be run for 1000 such cycles.

Writing codes:-

- The provided templates for the Client contain prewritten code to connect to the simulator.
- You will be writing your codes in a particular function called *process()*. Follow the instructions in each template and edit only the function specified.
- The function receives a string which stores the information the simulator sends, and an empty 'request' string to be populated by you.
- All the processing that needs to be done is done in that function and the request string that is to be sent back to the simulator is populated within that function.
- Please note: The above statement does not mean that you need to write all your code within the given function. You can write your own functions as well as define your own structures and classes if required. However, all your code should be within the specified region given in the template. You should be careful not to modify the main function or other provided methods/variables; otherwise your code will get disqualified.

Finally, your aim is to try and gather as many gold-packs and diamond-packs as possible using the 'Explorer' within the 1000 cycles.



Instructions

- The entire map is of dimension 150 x 150, bounded by virtual 'walls'. The Explorer dies if it crashes with the boundary walls.
- The map will have two types of regions:
 - i. Island: 30 x 30 at four corners, 60 x 60 at center.
 - ii. Foggy Zone: Belt of unit thickness around the central island.
 - iii. Ocean: Rest of the map.
- There will be Pirate ghosts guarding the islands. One ghost will guard each of the corner islands, and two ghosts will guard the center island.
- An additional ghost, the ghost of the Captain of the Pirate ship, will be patrolling the waters.
- The Explorer will start off from a random position, but not on any Gold or Diamond, and not on any of the islands. This starting position will be fixed for all the participants in the final round.
- The pirate ghosts start off from the outer vertices of the corner islands (bottom-right vertex of top-left island, bottom-left vertex of top-right island, top-right vertex of bottom-left island, top-left vertex of bottom-right island) and the top-left and bottom-right vertices of the center island. The captain ghost starts off from a random location on the water (also fixed for all the participants in the final round).
- Pirate ghost behavior is as follows:
 - i. Patrols boundary of the assigned islands when Explorer is not on that island.
 - ii. Chases Explorer if Explorer enters the island that it is guarding.
 - iii. Returns to its patrolling position if Explorer escapes the island / goes out of visibility.
- Captain ghost behavior is as follows:
 - i. Chases Explorer when Explorer is on ocean.
 - ii. Halts (freezes) when the Explorer leaves the waters / goes out of visibility.
- Both the Explorer and the ghosts can move only one step in each cycle, to any of the 8 surrounding squares. The pirate ghosts can only move within the assigned islands, and cannot leave them. Likewise, the captain ghost can never leave the water.
- When the Explorer enters the Foggy Zone, it goes invisible to the ghosts. It is a region where neither the land-guards (pirate ghosts) nor the water guard (captain ghost) can enter. The Explorer can move within the foggy zone undetected by the ghosts.
- 200 gold-packs (each worth 1 point) are spread throughout the ocean.
- 150 diamond-packs (each worth 5 points) are distributed throughout the islands in the following way:
 - i. 25 diamond-packs in each of the four corner islands.
 - ii. 50 diamond-packs in the center island.
- The foggy zone is barred of any treasure.
- The Explorer must step on a treasure-pack (gold/diamond unit) to grab it to add to its points. When an Explorer steals a gold-pack, 1 point is credited to him, and for a diamond-pack, 5 points are credited.
- The Explorer can 'buy' special Ghost-Buster grenades, each costing 10 points. Points are deducted automatically from the Explorer whenever he chooses to use a grenade (the total no. of points collected by the Explorer can never go negative, and hence, the Explorer must have atleast 10 points to its credit in order to use a grenade).
- When the Explorer tosses a grenade on a ghost, the ghost freezes for 5 cycles. The grenades can be thrown upto a maximum distance of 5 units (i.e., anywhere within an 11x11 square around the Explorer). In a cycle, the Explorer can either make a move or toss a grenade on to a square within its range. To be effective, the grenade must be pitched on to the exact position of a ghost (as on the previous cycle). It will be assumed that after being tossed, the grenade will take zero time to reach its destination & explode. If multiple ghosts are on the same square on which a grenade is pitched, it will affect both the ghosts in the same way.
- If a grenade is pitched on an already incapacitated ghost, the duration of effect of the previous grenade will overlap with that of the new one. Say, if a ghost is frozen due to a grenade thrown at it in $t=0$, and in the 3rd cycle of freeze ($t=2$), another grenade is tossed on to it, it will remain frozen for an additional 5 cycles (upto $t=7$).
- The Explorer dies if it is touched ('caught') by any of the ghosts (even when they are frozen).
- In every cycle, the Explorer makes the first move, and then the ghost. As such, if the Explorer swaps positions with a ghost in a cycle, it will be considered to have crashed into that ghost, and will be declared to have been 'caught'.
- Your code must receive the environment information from the server, process it and return the request string to the server within a server cycle (1 second). If the processing takes more than 1 second for any cycle, and no request is returned to the server within 1 second, the Explorer will be terminated.
- The objective is to maximize the points collected by Explorer at the end of 1000 server cycles. The total points collected by each participant's code will be recorded after 1000 Server cycles (or until the Explorer dies). The participant with the highest points wins. In case of a tie, the participant whose code (Explorer) is still alive will be preferred. If there is a tie on that too, the tied participants' codes will be given another run.

It should be interpreted as follows:

Explorer X-coordinate: 112

Explorer Y-coordinate: 89

25 x 25 Vision Matrix, with Explorer at the center:

```
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWGWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWGWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWGWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWGWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWGWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWGWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWGWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWGWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWGWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWGWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWGWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWGWWWW
LLLLWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWGWWWW
```

Rules:

- This is an online programming event, in which you will have to complete the codes in the template files that we will be providing. After completing those codes you will have to submit the completed code online.
- The templates will be in Java, C and C++. You can choose any of these programming platforms. For C/C++, separate templates are available for Windows & Linux platforms; you may use the one of your choice.
- Registration is compulsory to participate in this event.
- There can be a maximum of 2 members in a team.
- Write your code following the guidelines provided in the template. Any attempt of tampering with the template code will result in immediate disqualification.
- Make a zip archive named *sia<team-name>.zip* (e.g. *siaABC.zip*) containing the following:
 - i. The modified Client code.
 - ii. *Details.txt* containing name, college, stream, year and contact details of the participants.
 - iii. *Algorithm.txt* briefly describing the algorithm being used in your code.Send in the zip archive to **stealitaway@edg.co.in** before the submission deadline.
- In case of multiple submissions from the same team, only the last submission before the deadline will be considered.
- Your code must reach us by 11:59 pm 1st April, 2009. We will not be responsible for any mail bounces.

Downloads:Download the Simulator [Server] (for both Windows and Linux)

You will need JRE 1.5 (or higher) in order to run the Simulator. You can download JRE at java.sun.com. In order to run the simulator, try double-clicking it. If that does not work, open up the terminal / console / command-prompt, *cd* to the directory where you have downloaded the Server and type *java -jar Server.jar*. Make sure you start this simulator before running your client-side code.

You may choose from any of the following Templates for your code:

- Download Java Template [Client] (for both Windows and Linux)
You will need JDK in order to compile and run this Template. You can download JDK at java.sun.com.
- Download C/C++ Template [Client] (for Windows)
You will need Visual C++ 6.0 (or higher) to compile and execute this Client Template. Unzip the archive and use the 'Client.dsw' file to open the workspace. You may write your code either in C or in C++ within this template.
- Download C/C++ Template [Client] (for Linux)
Compile using *g++ Client.cpp*. No libraries need to be linked to this template. If you need to link any standard libraries to your code, please mention the compilation command in your mail. You may write your code either in C or in C++ within this template.

Download the Instructions file (PDF)

This file contains all the information given on this web page.

Prize Money:

- 1st Prize: Rs. 3,000
- 2nd Prize: Rs. 2,000
- 3rd Prize: Rs. 1,000

Coordinator:

Kaustav Dey Biswas – stealitaway@edg.co.in (09830964772)