Zadanie: RPG
RPG

Olimpiada Informatyczna

ONTAK 2013, dzień 7. Plik źródłowy `rpg.*` Dostępna pamięć: 64 MB. *15.08.2013*

Bajtomir oraz Bajtosław grają w grę typu RPG. Właśnie pokonali oni razem silnego potwora i z tego tytułu uzyskali wiele przedmiotów do podziału. Każdy przedmiot wpływa na dwie cechy bohaterów: magię (tj. umiejętności merytoryczne) oraz umiejętności motoryczne. Magia może być biała lub czarna. W grze występują dwie umiejętności motoryczne: latanie i pływanie. Początkowe umiejętności obu graczy w każdej z tych dziedzin są zerowe.

Każdy przedmiot opisują dwie liczby: wpływ na magię oraz wpływ na umiejętności motoryczne. Jeżeli liczba opisująca wpływ na magię jest dodatnia, to zwiększa umiejętności bohatera w używaniu białej magii, a jeżeli ujemna, to zwiększa umiejętności w używaniu czarnej magii. Bohater nie może być jednocześnie dobry i zły, tzn. jeżeli zdobędzie on przedmioty o dodatnim wpływie na białą magię, a potem przedmioty o takim samym wpływie na czarną magię, to nie będzie umiał żadnej z nich. Sprawa wygląda podobnie przy wpływie na umiejętności motoryczne; bohater nie może naraz rozwijać umiejętności latania i pływania.

Im większy uzyskany wpływ na jakąś cechę, tym coraz większe korzyści z uzyskanych umiejętności — jeżeli któryś bohater uzyska mały wpływ na magię, to nie pokona dzięki niej prawie nikogo, natomiast im więcej tej magii, tym silniejszych przeciwników będzie mógł pokonać. Dlatego jeżeli bohater uzyska łączny wpływ a na jakąś umiejętność, to zysk doświadczenia z tego tytułu jest określony jako a^2 (zauważ, że a może być zarówno dodatnie, jak i ujemne). Dlatego jeżeli któryś z bohaterów uzyska łącznie wpływ na odpowiednie cechy równy a i b, to zysk doświadczenia dla niego wynosi $a^2 + b^2$.

Teraz przyszedł moment podziału uzyskanych przedmiotów. Gracze będą je wybierać na zmianę i muszą oni razem wziąć wszystkie uzyskane przedmioty. Każdemu z graczy zależy przede wszystkim na tym, aby być silniejszym od tego drugiego. W szczególności, nie jest dla nich ważne to, ile łącznie zyskają dzięki zdobytym przedmiotom, tylko to, żeby uzyskać więcej niż drugi gracz. Jako że Bajtomir wykazał się większą odwagą w pokonywaniu potwora, ma on prawo wybrać, który z graczy będzie pierwszy wybierać swój przedmiot. Pomóż mu i powiedz, czy powinien on zaczynać oraz które przedmioty powinien wybierać, aby zyskać więcej doświadczenia z tytułu ich uzyskania!

Komunikacja programu

Twój program nie powinien nic czytać z wejścia ani wypisywać na wyjście. Zamiast tego zostanie on skompilowany z biblioteką oceniającą, która będzie symulować Bajtosława grającego z Tobą. Na początku programu umieść dyrektywę:

```
#include "crpglib.h"
```

Biblioteka udostępnia cztery funkcje:

- `vector<pair<int, int> > inicjuj()` — funkcję tę Twój program powinien wywołać raz, na początku działania programu. Zwraca ona wektor par oznaczających wpływ poszczególnych przedmiotów na magię i na umiejętności motoryczne. Przedmioty są ponumerowane od 1 do n w takiej kolejności, w jakiej zostały podane w tym wektorze, przy czym n to liczba przedmiotów ($1 \leq n \leq 100\,000$). Wpływy na każdą cechę nie będą przekraczać 10 000 co do wartości bezwzględnej.
- `void wybierz(bool b)` — funkcja ta powinna zostać wywołana dokładnie raz. Wywołanie to ma nastąpić po wywołaniu funkcji `inicjuj()` oraz przed wszystkimi wywoływaniami funkcji `wez(k)` oraz `przeciwnik()`. Jeżeli Bajtomir ma rozpoczynać rozgrywkę, to argumentem funkcji powinno być **true**, a w przeciwnym przypadku — **false**.
- `void wez(int k)` — wywołanie tej funkcji z argumentem k oznacza, że chcemy, aby Bajtomir wziął przedmiot o numerze k.
- `int przeciwnik()` — wywołanie tej funkcji zwraca numer przedmiotu, który został wybrany przez Bajtosława.

Funkcje `wez(k)` oraz `przeciwnik()` muszą być wywoływane naprzemiennie. Ich łączna liczba wywołań musi być równa liczbie przedmiotów.

Ocenianie

Twój program dostanie punkty za dany test wtedy i tylko wtedy, gdy wzięte przez niego przedmioty dadzą Bajtomirowi więcej zysku doświadczenia niż Bajtosławowi. Można założyć, że dla każdego testu jest to możliwe.

W testach wartych 30% punktów przedmioty będą miały zerowy wpływ na umiejętności motoryczne.

Przykładowa biblioteka

Przykładowy program wykorzystujący bibliotekę znajduje się w pliku `rpgzaw.cpp`. Do Twojej dyspozycji jest również przykładowa biblioteka `crpglib.cpp`, która implementuje podane cztery funkcje. `inicjuj()` wczytuje listę przedmiotów ze standardowego wejścia. Na wejściu w pierwszym wierszu powinna znajdować się liczba przedmiotów n, w następnych n wierszach po parze liczb a i b określających wpływ danego przedmiotu.

Zarówno biblioteka jak i program grają nieoptymalnie, wybierają niewykorzystany jeszcze przedmiot o najmniejszym numerze.

Aby przetestować swój program z tą biblioteką, wpisz komendę:

```
g++ program.cpp crpglib.cpp -o rpg -O2
```

a następnie

```
./rpg < Twoja_lista_przedmiotow.txt
```