Zadanie 6: Wyszukiwanie

Otwarto: poniedziałek, 15 stycznia 2024, 10:00
Wymagane do: środa, 24 stycznia 2024, 16:00

Uwaga: Osoby, które chciałyby być zwolnione z egzaminu i chcą w tym celu wykorzystać wynik z tego zadania, zachęcamy, żeby oddały swoje rozwiązanie przed swoim ostatnim laboratorium.

Ala wymyśliła sobie pewną liczbę x z zakresu od 1 do n. Bolek próbuje zgadnąć tę liczbę. Jedyne pytania, jakie może zadawać Bob, są postaci: "Czy x < y?" dla dowolnie wybranych przez niego liczb y. Na każde z takich pytań Bolek otrzymuje odpowiedź "Tak" lub "Nie". Zadaniem Bolka jest odgadnąć x, zadając jak najmniej pytań. Niestety Ala może czasem kłamać. Twoim zadaniem jest napisanie programu, który gra w tę grę jako Bolek.

Rolę Ali odgrywa biblioteka, której interfejs otrzymujesz w pliku wys.h. Na początku interakcji z biblioteką powinieneś wywołać funkcję

```
void dajParametry(int &n, int &k, int &g);
```

Funkcja podaje Ci trzy liczby: liczbę `n` określającą zakres, liczbę `k` oznaczającą maksymalną liczbę razy, ile Ala może skłamać w jednej grze, oraz liczbę `g` oznaczającą liczbę gier do rozegrania przy danych parametrach `n` oraz `k`. Od wywołania tej funkcji rozpoczyna się pierwsza gra.

W każdej z gier możesz zadawać Ali pytania jedynie za pomocą funkcji

```
bool mniejszaNiz(int y);
```

Wynikiem funkcji jest odpowiedź Ali na pytanie "Czy x < y?", dla nieznanej Ci liczby x. Dokładniej, jeśli wynikiem funkcji jest `true`, oznacza to, że Ala odpowiada "Tak", a jeśli wynikiem jest `false`, to Ala odpowiada "Nie". Odpowiedź Ali nie musi być prawdziwa, jednak w ciągu jednej gry Ala może skłamać co najwyżej k razy.

Na końcu gry powinieneś podać Ali swoją odpowiedź za pomocą funkcji

```
void odpowiedz(int x);
```

Wywołanie tej funkcji powoduje automatycznie rozpoczęcie kolejnej gry, jeśli jeszcze nie zostało rozegranych g gier. W przeciwnym razie wywołanie funkcji zakończy działanie Twojego programu.

Twój program powinien zadawać minimalną możliwą liczbę zapytań w następującym sensie: Jeśli dla danych n oraz k Bob może zgadnąć odpowiedź za pomocą z pytań niezależnie od tego, jaka jest wartość x oraz kiedy Ala zdecyduje się kłamać (jeśli w ogóle), to Twój program dla tych n oraz k musi zawsze udzielać poprawnej odpowiedzi za pomocą co najwyżej z pytań.

Twój program powinien działać dla parametrów: $1 \le n \le 12$, $0 \le k \le 3$, $1 \le g \le 10.000$. Czas działania programu nie powinien mieć znaczenia, o ile nie będzie przesadnie zły (kilka sekund będzie OK).

Przykładową strategię Ali znajdziesz w pliku wyslib.cpp. Przykładowe, nieoptymalne rozwiązanie implementujące strategię Boba możesz znaleźć w pliku wys_naive.cpp. (**UWAGA:** rozwiązanie to zostało lekko poprawione.) Rozwiązanie to możesz skompilować wraz z podaną biblioteką za pomocą poniższej komendy, przy takich samych opcjach kompilacji jak w poprzednich dwóch zadaniach:

```
g++ @opcje wys_naive.cpp wyslib.cpp -o wys_naive
```

wys_naive.cpp	17 stycznia 2024, 14:45
wys.h	14 stycznia 2024, 23:37
wyslib.cpp	14 stycznia 2024, 23:37

?