

Ćwiczenia w stosowaniu techniki zachłannej i programowania dynamicznego.

Rozwiązanie każdego z zadań polega na:

- 1) zaproponowaniu możliwie najefektywniejszego rozwiązania opartego na programowaniu dynamicznym lub algorytmie zachłannym
- 2) określeniu pesymistycznej złożoności czasowej rozwiązania

Problem 1 - Górski hotel

Pewna alpejska miejscowość jest znana ze swojego położenia w otoczeniu wspaniałych gór. Turyści z całego świata przyjeżdżają podziwiać tutaj pionowy świat pełen skalnych ścian i wodospadów. Można tam spotkać przedstawicieli różnych narodowości: od Amerykanów po wyjątkowo licznych Chińczyków. Pewien mały okoliczny hotelik (dysponujący dwoma apartamentami) ma wyjątkowe powodzenie wśród przyjezdnych – położony jest on nad samą przepaścią, a z jego okien otwierają się bajkowe widoki. Mimo wielkich cen nie brakuje chętnych na noclegi i wschód słońca z widokiem na białe szczyty. Ze względu na dużą popularność miejsca na świecie właściciele hotelu postanowili zmienić strategię rezerwacji i maksymalnie zwiększyć swoje zyski (góry górami, ale kasa musi się zgadzać). Hotel przyjmuje rezerwacje od gości z dużym wyprzedzeniem, ale bez ustalonej ceny – każdy deklaruje ile jest w stanie zapłacić za pobyt w deklarowanym okresie. Po zebraniu zgłoszeń właściciele hotelu sami decydują, komu przyznać pokoje – kryterium jest oczywiście jak największy zysk. Okazało się jednak, że rezerwacji jest tyle, że bez pomocy komputera zadanie wyselekcjonowania najlepszych zgłoszeń jest niemożliwe. Zdenerwowany właściciel (a miało być tak pięknie!) pilnie poszukuje informatyka, który przygotuje odpowiednie oprogramowanie. Za usługę na pewno zapłaci hojnie (po szwajcarsku). Wiadomo, że pojedyncza rezerwacja zawsze obejmuje jeden apartament na zadany okres czasu. Dwoch rezerwacji nie można przydzielić do tego samego apartamentu jeżeli zachodzą na siebie terminami. Ponadto, koniec rezerwacji apartamentu w danym dniu oznacza, że od tego dnia można tam przyjmować gości w ramach kolejnej rezerwacji. Pomóż zarobić hotelowi jak najwięcej, a nie pożałujesz.

Wejście

W pierwszej linii wejścia pojawia się liczba całkowita n ($1 \leq n < 300$) oznaczająca liczbę rezerwacji. W kolejnych n liniach podane są informacje o kolejnych rezerwacjach: w każdej linii po trzy liczby całkowite p , k oraz z oddzielone spacjami ($1 \leq p \leq 300$, $p < k \leq 300$, $1 \leq z \leq 1000$) oznaczające odpowiednio dzień początku rezerwacji, dzień końca rezerwacji oraz zysk z tej rezerwacji.

UWAGA: Na wejściu nie istnieje taka para rezerwacji, która kończy się tego samego dnia.

Wyjście

W jednym wierszu wyjścia ma się pojawić jedna liczba całkowita oznaczająca maksymalny przychód jaki hotel może uzyskać z realizacji rezerwacji.

Przykład

Wejście:

```
5           //5 rezerwacji
9 11 2      //pierwsza rezerwacja od dnia 9 do dnia 11 (zapłata 2)
1 5 4
1 8 7
5 9 4
6 10 5
```

Wyjście:

```
18          //zysk 18 (apartament nr 1: rezerwacje nr 1 i 3, apartament nr 2: rezerwacje nr 2 i 5)
```

Problem 2 – Napęd

Jest XXV wiek, a w naszej galaktyce toczą się konflikty różnych cywilizacji. Statek Gwiazdnej Floty ma do wykonania ważną misję, ale w wyniku ataku stracił on paliwo do napędu nadprzestrzennego. Bez możliwości podróżowania z prędkością większą niż światło ludzie nie mają szans dolecieć na czas do celu. Na planecie Deneb V, gdzie obecnie się znajdują, jest możliwość nabycia wielkich kryształów, z których być może da się szybko uzyskać odpowiednią mieszankę paliwową. Paliwo musi posiadać w składzie trzy substancje: X, Y oraz Z w proporcji 1:1:1. Naruszenie tych proporcji może spowodować przegrzanie reaktorów napędu i eksplozję statku. Niestety same kryształy nie są oczyszczone i każdy z nich zawiera pewne ilości wszystkich trzech składników. Proces rozdzielania składników jest bardzo czasochłonny, a misja nie cierpi żadnej zwłoki. Stąd załódze zależy na dobraniu takiej grupy kryształów, dla której sumaryczne masy substancji X, Y oraz Z będą spełniały pożądaną proporcję. Oczywiście najlepiej, by z takiej proporcji dało się wytworzyć jak najwięcej paliwa (jednak nie więcej niż 300 ton – tyle ładunku mogą zabrać). Dodatkowo, każdy kryształ ma pewną (jednakową) cenę, a więc najlepiej byłoby osiągnąć cel ponosząc jak najmniejsze koszty. Mając dane składu poszczególnych kryształów pomóż kapitanowi znaleźć optymalne rozwiązanie.

Wejście:

W pierwszej linii wejścia podana jest liczba całkowita n kryształów ($1 \leq n \leq 200$). W kolejnych n liniach znajdują się informacje o kryształach. W każdej z nich są trzy liczby całkowite określające odpowiednio ilość (w tonach) substancji X, Y oraz Z w danym kryształcie.

Wyjście:

W pierwszej linii wyjścia ma być podana maksymalna ilość paliwa (ale nie większa niż 300 ton), którą da się uzyskać z pewnej grupy kryształów. W kolejnej linii ma się pojawić minimalna liczba kryształów potrzebnych do tego celu. Jeśli nie da się uzyskać paliwa, to program ma wypisać NIE.

Przykład:

Wejście:

```
5 //dostępnych do kupna jest 5 kryształów
1 2 3 //pierwszy kryształ zawiera 1 tonę substancji X, 2 tony substancji Y i 3 tony substancji Z
1 1 1 //itd.
1 0 0
1 1 0
2 1 0
```

Wyjście:

```
12 //Da się uzyskać maksymalnie 12 ton paliwa (potrzebne po 4 tony każdej substancji)
3 //Do tego celu wystarczy kupić tylko 3 kryształy (pierwszy, drugi i piąty)
```

Problem 3 - Rzymscy najemnicy

Pewien Rzymianin żył sobie w szczęściu i bogactwie. Pewnego dnia stało się nieszczęście, gdyż jego ukochana została uprowadzona przez jego zacieklých wrogów. Co mu teraz po samym bogactwie, gdy szczęścia nie ma? A jednak pieniądze mogą mu pomóc – ma on zamiar wynająć całą drużynę najemnych żołnierzy, którzy pomogą mu odbić ukochaną. Sprawa nie jest jednak taka prosta, gdyż rzymscy najemnicy się wysoko cenią. Każdy z nich ma swoje wymagania w czasie misji. Zgodnie z powiedzeniem „chleba i igrzysk” żołnierze oczekują prowiantu i rozrywek w zamian za świadczone przez nich usługi. Jeśli którykolwiek z tych warunków nie zostanie spełniony, to najemnik odchodzi w poszukiwaniu lepszego pracodawcy, a nasz Rzymianin dalej wzdycha do ukochanej. Ma on pewien górny próg na sumę prowiantu i sumaryczną rozrywkę, jaką może zapewnić dla drużyny najemnej. Ograniczenia wynikają zarówno z budżetu jak i z logistyki. Mieszcząc się w tych ograniczeniach nasz bohater chce skompletować oddział o możliwie największej sile rażenia. A może Ty pomożesz mu w tym zadaniu?

Wejście:

W pierwszej linii wejścia podane są liczby P i R ($1 \leq P, R \leq 1000$) będące maksymalnymi ilościami prowiantu i rozrywki, które może zapewnić Rzymianin. W drugiej linii podana jest liczba n ($1 \leq n \leq 100$) najemników. W kolejnych n liniach podane są liczby s_i, p_i, r_i ($0 \leq s_i, p_i, r_i \leq 100000$) oznaczające odpowiednio: siłę i -go żołnierza, jego wymagania co do prowiantu oraz rozrywek.

Wyjście:

W pierwszej linii wyjścia ma być podana sumaryczna siła najmocniejszego oddziału leżącego w możliwościach wynajmu Rzymianina. W kolejnej linii podane są numery wynajętych żołnierzy (numeracja od 1).

Przykład:

Wejście:

```
10 8      //Rzymianin może zapewnić 10 jednostek prowiantu i 8 rozrywek
4         //Jest 4 najemników
3 5 2     //Pierwszy z nich ma siłę 3, wymaga 5 prowiantu i 2 rozrywki
4 6 3     //itd.
2 4 6
3 5 9
```

Wyjście:

```
5         //najmocniejszy wynajęty oddział może mieć siłę 5
1 3       //składa on się z żołnierzy nr 1 oraz 3, Rzymianin może sprostać ich wymaganiom
```