**Семінар-практикум: «Методичні принципи підготовки школярів до олімпіад, конкурсів, турнірів. Практикум з розв’язування задач в Python»**

**e-olimp**

<https://www.e-olymp.com/uk/problems/2392>

**Цікава сума**

Задано тризначне натуральне число **n**. Визначити суму найбільшого та найменшого трицифрових чисел, які можуть бути утворені з числа **n** перестановкою цифр.

**Вхідні дані**

Натуральне число **n** (**100** ≤ **n** ≤ **999**).

**Вихідні дані**

Вивести суму найбільшого та найменшого трицифрового числа.

Ліміт часу **1** секунда

функцию sorted()

reverse=True

метод list.sort()

s=''.join(a)

Ліміт використання пам'яті **128** MiB

**Вхідні дані #1**

524

**Вихідні дані #1**

787

|  |  |  |  |
| --- | --- | --- | --- |
| n=input()print(n)a=list(n)print(a)a=sorted(a)print(a)x=''.join(a)print(x)a=sorted(a,reverse=True)print(a)y=''.join(a)print(y)print(int(x)+int(y)) | n=input()a=sorted(n)b=sorted(a,reverse=True)x=''.join(a)y=''.join(b)print(int(x)+int(y)) | n=input()a=sorted(n)b=sorted(a,reverse=True)x=''.join(a)y=''.join(b)if int(x)<10: x=yprint(int(x)+int(y)) | n=input()a=sorted(n)b=sorted(a,reverse=True)x=''.join(a)y=''.join(b)if int(x)<10: x=yelif int(x)<100:  a[0],a[1]=a[1],a[0] x=''.join(a)print(int(x)+int(y)) |

**All-Ukrainian Collegiate Programming Contest
I stage
September 26, 2020**

**Задача H. Зменшити масив**Дано *n* цiлих чисел *a*1*; a*2*; : : : ; an*. Кожне число можна не бiльше одного разу зменшити на 1.
Перевiрте, чи можна зробити масив неспадаючим, тобто *ai* ≤ *ai*+1 для кожного *i* (1 ≤ *i < n*).
**Формат вхiдних даних**Перший рядок мiстить одне цiле число *n* (1 ≤ *n* ≤ 105) — кiлькiсть чисел.
Другий рядок мiстить *n* цiлих чисел *a*1*; a*2*; : : : ; an* (1 ≤ *ai* ≤ 109).
**Формат вихiдних даних**Виведiть «Yes», якщо це можливо, або «No», якщо нi.
Кожну букву можна виводити у будь-якому регiстрi.
**Приклади**

|  |  |
| --- | --- |
| standard input  | standard output |
| 51 2 1 1 3 | Yes |
| 41 3 2 1 | No |
| 51 2 3 4 5 | Yes |
| 11000000000 | Yes |

|  |  |
| --- | --- |
| n=int(input())a=list(map(int,input().split()))d=1f=Truefor i in range(n-1): print(i,a[i],a[i+1]) if a[i]<=a[i+1]: if d>=1: a[i]-=1 elif a[i]-a[i+1]==1 and d>=1: a[i]-=1 else: f=False break d=a[i+1]-a[i] print(a)if f: print("Yes")else: print("No") |  |

**Київська мiська iнтернет-олiмпiада з iнформатики 2020 - I тур**

**Задача E: Перевiрка числа**У Козака Вуса є улюблена цифра — *d*.
Йому подарували число *n*. Перевiрте, чи у цьому числi є цифра *d*.
**Формат вхiдних даних**Перший рядок мiстить два цiлi числа *n* та *d* (0 ≤ *n ≤* 999, 0 ≤ *d ≤* 9).
**Формат вихiдних даних**Виведiть “Yes”, якщо в *n* є цифра *d*, або “No” iнакше.
Букви можна виводити у будь-якому регiстрi. Тобто, якщо вiдповiдь “Yes”, то можна вивести,
наприклад, “YES”, або “yEs”.
**Приклади**

|  |  |
| --- | --- |
| standard input  | standard output |
| 394 4  | Yes |
| 111 3  | No |

|  |  |
| --- | --- |
| n,d=map(str,input().split())if n.find(d)!=-1: print("Yes")else: print("No") |  |

**Задача D: Паркування**Козак Вус приїхав у торговий центр. Перед тим, як пiти купити собi новi шаровари, йому потрiбно припаркуватися. У торговому центрi є три тарифи для оплати паркування:
1. Можна за кожну хвилину перебування на стоянцi платити *a* гривень.
2. Можна за *k* хвилин заплатити *b* гривень. Звернiть увагу, що не обов’язково використовувати
всi *k* хвилин.
3. Можна заплатити *c* гривень та перебувати на стоянцi стiльки, скiльки потрiбно.
Козак знає, що вiн буде у торговому центрi рiвно *t* хвилин. Вус хоче вибрати такий тариф (чи
кiлька тарифiв), щоб заплатити як можна менше. Допоможiть йому знайти мiнiмальну суму, яку
йому потрiбно заплатити.
**Формат вхiдних даних**Перший рядок мiстить одне цiле число *t* (1 ≤ *t ≤* 1 000) — кiлькiсть хвилин, протягом яких
Козак Вус буде на стоянцi.
Другий рядок мiстить одне цiле число *a* (1 ≤ *a ≤* 1 000) — цiна за одну хвилину.
Третiй рядок мiстить два цiлi числа *k* та *b* (1 ≤ *k; b ≤* 1 000), якi означають, що за *k* хвилин
можна заплатити *b* гривень.
Четвертий рядок мiстить одне цiле число *c* (1 ≤ *c ≤* 1 000 000) — цiна за перебування на стоянцi
протягом довiльного часу.
**Формат вихiдних даних**Виведiть одне цiле число — вiдповiдь на задачу.
**Приклад**

|  |  |
| --- | --- |
| standard input  | standard output |
| 1013 2100 | 7 |
| 1123 2100 |  |

**Примiтка**Козак Вус може використати другий тариф тричi, заплативши ≤ гривень та отримавши 9 хвилин.
Ще одну хвилину вiн може купити за допомогою першого тарифу за 1 гривню.

|  |  |
| --- | --- |
| t=int(input())a=int(input())k,b=map(int,input().split())c=int(input())v1=t\*av2=(t+k-1)//k\*bv3=t//k\*b+t%k\*av4=cprint(min(v1,v2,v3,v4)) |  |

<https://www.e-olymp.com/uk/problems/7818>

# ПОЇЗДКИ на МЕТРО

Жетон для однієї поїздки на метро коштує **4** грн. Якщо планується більше поїздок, то врешті, можна придбати проїзний квиток, ціна якого на **10** поїздок становить **30** грн., на **50** – **125** грн.,а на **100** – **200** грн. Яку мінімальну суму потрібно витратити, щоб оплатити проїзд не менш, як **M** поїздок на метро?

#### Вхідні дані:

Натуральне число **M**. **1** ≤ **M** ≤ **1000**.

#### Вихідні дані:

Відповідь до задачі.

**Вхідні дані #1**content\_copy

25

**Вихідні дані #1**content\_copy

80

|  |  |
| --- | --- |
| m=int(input())x200=((m+99)//100)\*200x125=(m//100)\*200+((m%100+49)//50)\*125x30=(m//100)\*200+(m%100//50)\*125+((m%50+9)//10)\*30x4=(m//100)\*200+(m%100//50)\*125+(m%50//10)\*30+m%10\*4print(min(x200,x125,x30,x4)) |  |

**<https://new.netoi.org.ua/>**

**Задача Detour.**  Поле в «клітинку» розмірами n\*m потрібно обійти по спіралі (так як це  показано на малюнку), відвідуючи кожну клітинку. Починати обхід слід з верхньої лівої клітинки. Скільки поворотів (зрозуміло направо на 90 градусів) слід виконати?

**Технічні умови.** Програма **Detour** читає з пристрою стандартного введення два числа через пропуск ***n*, *m* (1≤*n*, *m* ≤500000).** Програма виводить одне ціле число – кількість поворотів.

**Приклад**

|  |  |
| --- | --- |
| *Введення* | *Виведення* |
| 3 5 | 4 |

 Пояснення

|  |  |  |  |
| --- | --- | --- | --- |
| n | m | R |  |
| висота | ширина |  |  |
| 1 | 1 | 0 | - |
| 1 | 100 | 0 | - |
| 100 | 1 | 1 | | |
| 2 | 2 | 2 | - |
| 2 | 100 | 2 | - |
| 100 | 2 | 3 | | |
| 3 | 3 | 4 | - |
| 3 | 100 | 4 | - |
| 100 | 3 | 5 | | |
|  4 | 4 | 6 | - |
| 4 | 100 | 6 | - |
| 100 | 4 | 7 | | |
| 5 | 5 | 8 | - |
| 5 | 100 | 8 | - |
| 100 | 5 | 9 | | |
| 6 | 6 | 10 | - |
| 6 | 100 | 10 | - |
| 100 | 6 | 11 | | |
| 7 | 7 | 12 | - |
| 7 | 100 | 12 | - |
| 100 | 7 | 13 | | |
| 8 | 8 | 14 | - |
| 8 | 100 | 14 | - |
| 100 | 8 | 15 | | |

|  |  |
| --- | --- |
| C++ | python |
| ##include <iostream>#include "math.h"using namespace std;int main(){ long long n,m,r; cin>>n>>m; r=(min(n,m)-1)\*2;if (min(n,m)==m && m<n)r++;cout << r << endl; return 0;} | import matha,b=map(int,input().split())r=(min(a,b)-1)\*2if min(a,b)==b and b<a: r+=1print(r) |



<https://www.e-olymp.com/uk/problems/1619>

Пограбування будинків

Ви - професіонал своєї справи і плануєте пограбувати ряд будинків уздовж вулиці. У кожному будинку захована певна сума грошей. Єдине, що заважає Вам грабувати - так це те, що сусідні будинки пов'язані системою безпеки: буде переданий сигнал в поліцію, якщо два сусідні будинки будуть пограбовані в один і той же вечір.

Знаючи кількість грошей в кожному будинку, визначте максимальну суму, яку Ви зможете пограбувати сьогодні ввечері без попередження поліції.

**Вхідні дані**

Перший рядок містить кількість будинків **n** (\*\*1 ≤ n ≤ **106**\*\*). Другий рядок містить **n** цілих невід'ємних чисел **a1,a2**, ...,**an**, де **ai**- кількість грошей, яке може бути винесено з **i** - го будинку.

**Вихідні дані**

Виведіть максимальну суму, яку Ви зможете пограбувати сьогодні ввечері без надходження сигналу в поліцію.

Ліміт часу **3** секунда

Ліміт використання пам'яті **128** MiB

**Вхідні дані #1**

5

6 1 2 10 4

**Вихідні дані #1**

16

|  |  |
| --- | --- |
| n=int(input())a=list(map(int,input().split()))a.append(0)a.append(0)a[2]=a[2]+a[0]for i in range(3,n): a[i]=max(a[i - 2], a[i - 3]) + a[i] print(max(a)) | **n**=int(**input**())a=**list**(map(int,**input**().**split**()))a.append(0)a.append(0)b=[0]\*(**n+2)**b[0] = a[0];b[1] = a[1];b[2] = b[0] + a[2];**for** i **in** **range**(3,**n**): b[i] = max(b[i - 2], b[i - 3]) + a[i];**print**(max(b[**n** - 1], b[**n** - 2])) |

[Папка з матеріалами](https://drive.google.com/drive/folders/1EuMyxJCDpdvtUpSXeYsdIhjJCD9rcRMx?usp=sharing)