**1 тур Волинської учнівської оілмпіади з програмування**

**Задача 1. Число з однакових цифр**

Нехай в калькулятор введемо деяке натуральне число N. Натиснемо клавішу +. Ваше завдання: отримати на екрані число яке складається з однакових цифр. Для цього можна виконувати тільки одну дію: натискувати клавішу = (можливо і 0 раз). Після першого натискування отримаємо результат N+N, після наступного натискування результат збільшується на N. Потрібно визначити, чи можна виконати це завдання. Якщо можна, то визначити перше число яке складається з однакових цифр. Кількість цифр які може відображати калькулятор вважати необмеженою.

|  |  |
| --- | --- |
| #include "fstream"  using namespace std;  ifstream cin("input.txt");  ofstream cout("output.txt");  int main()  {long long n,i,b,c,k,j,t;  cin>>n;  long long f=0,s=0;  int x[1000];  while(f==0 && s<=10000000){  s=s+n;  t=s;i=0;  while(t>0){  i++;  x[i]=t%10;  t=t/10;  }  b=0;  //for(j=1; j<=i; j++)cout<<x[j]<<" ";cout<<endl;  for(j=1; j<i; j++){  if(x[j]!=x[j+1])b=1;  }  if(b==0){  f=1;  c=x[1]; k=i;}    }  if (f==1) cout<<c<<" "<<k<<endl;  else cout<<"Impossible"<<endl;  return 0;  } | #include <fstream>  using namespace std;  int main()  {  ifstream cin("input.txt");  ofstream cout("output.txt");  unsigned long long s,n,temp;  long i1,c1,c2,k; int f=0;  cin>>n;  s=n;  do  {f=0;  temp=s;  c2=temp%10;  k=0;  while (temp>0 && f==0)  {c1=temp%10;  k++;  //cout<<c1<<" "<<k<<endl;  if(c1!=c2)f=1;  temp=temp/10;  c2=c1;  }  if (f==0) f=2;  s=s+n;  }  while (f!=2 && s<1000000000);  if (f==2) cout<<c1<<" "<<k<<endl;  else cout<< "Impossible"<<endl;  return 0;  } |
| #include <fstream>  #include <math.h>  using namespace std;  int main()  {  ifstream cin("input.txt");  ofstream cout("output.txt");  unsigned long long s[10],n,temp;  long i,c,k;  int f=0;  cin>>n;  k=0;  for (i=1;i<=9;i++)s[i]=0;  while (f==0 && s[9]<17446744073709551615)  {  for (i=1;i<=9;i++)  {  s[i]=s[i]\*10+i;  if (s[i]%n==0){f=1;c=i;break;}  // cout<<s[i]<<"------------"<<i<<"---"<<k<<endl;  }  k++;  }  if (f==1) cout<<c<<" "<<k<<endl;  else cout<< "Impossible"<<endl;  return 0;  } | #include "fstream"  using namespace std;  ifstream cin("input.txt");  ofstream cout("output.txt");  int a[10][2001], f, n, i,k, x, y, j,st;  int main()  {  cin>>n;  f=0,i=0;  while(f==0 && i<=1000)  {  i++;  for(j=1; j<=9; j++)  { a[j][i]=j;  st=0;  for(k=1;k<=i;k++)st=(st\*10+a[j][k])%n;  if(st==0){x=a[j][1]; y=i;f=1;break;}  }  }  if (f==1)  cout<<x<<" "<<y<<endl;  else cout<<"Impossible"<<endl;  return 0;  } |

**Задача 2. I-й шлях у таблиці**

Задано таблицю N\*M, у кожній клітинці якої написана одна рядкова буква англійського алфавіту. Розглянемо всі можливі шляхи від лівого верхнього кутка до правого нижнього кутка, якщо вам дозволено йти тільки вправо і вниз. Конкатенація букв у процесі руху утворює рядок. Будемо вважати, що цей рядок – значення шляху. Тепер розглянемо всі такі шляхи і відсортуємо їх значення в алфавітному порядку. Ваше завдання: знайти значення I-го шляху в цьому відсортованому списку.

|  |  |
| --- | --- |
| void p(int k,int i0,int j0)  {int x,y;  c[k]=a[i0][j0];  if(i0==n-1 && j0==m-1){  if(ii==nomer-1){  for(int i=0;i<n+m-1;i++){cout<<c[i];}  cout<<endl;  }  ii++;  }  else  {  if(i0<n-1)p(k+1,i0+1,j0);  if(j0<m-1 )p(k+1,i0,j0+1);  } |  |

|  |
| --- |
| **Завдання 1 туру NetOI-2015 (16.10-11.11) 2015 р.** |
| |  | | --- | |  |  |  | | --- | | Розв'язки приймаються до   0 годин 12 листопада 2015 р. | |