

ЗИМНИ МАТЕМАТИЧЕСКИ ПРАЗНИЦИ – РУСЕ’ 2006

```
int j = 1;
for (; i <= d; i++)
{
    printf("%d ", j);
    j += s + 1;
    if (i >= d - 1)
    {
        j++;
    }
}
printf("\n");

return 0;
}
```

Задача D3. MAX3

Зададени са **N** цели числа. Напишете програма **MAX3**, която избира три от тях, произведението на които е максимално.

Вход: Данните се четат от стандартния вход. На първия ред е разположено числото **N** – брой на числата от последователността ($3 \leq N \leq 10^6$). На следващите редове са разположени и самите числа, които по абсолютна стойност не надхвърлят 30 000.

Изход: На стандартния изход се извеждат три числа, произведението на които е най-голямо. Числата се извеждат в произволен ред. Ако съществуват няколко различни тройки числа, които дават максимално произведение, се извежда коя да е от тях.

ПРИМЕР:

Вход:	Изход:
9 3 5 1 7 9 0 9 -3 10	9 10 9
3 -5 -30000 -12	-5 -30000 -12

Решение:

Ясно е, че поради големия брой на числата, тривиалният подход за сравняване на произведенията на всички възможни тройки числа не е добро решение, защото е много бавно.

Затова е необходимо да се приложи по-добър алгоритъм. Ако последователността съдържа само неотрицателни числа, то е ясно, че максималното произведение ще се получи от трите най-големи числа в нея. Ще ги означим с $max1$, $max2$ и $max3$.

Но в нашия случай последователността може да съдържа и отрицателни числа, което изисква по-подробен анализ. Произведението на две отрицателни числа е положително и затова е необходимо да намерим двете най-малки отрицателни числа - $min1$ и $min2$. Тогава максимално произведение ще получим като умножим $max1$ с по-голямото от $max2*max3$ или $min1*min2$. Тъй като произведението на три цели числа може да стане много голямо, за да не се получи препълване (самото произведение в задачата не се търси), удачно е да се сравняват най-напред $max2*max3$ и $min1*min2$.

Ако последователността се състои само от отрицателни числа, то максималното произведение (отрицателно) ще се получи отново от $max1$, $max2$ и $max3$.

ЗИМНИ МАТЕМАТИЧЕСКИ ПРАЗНИЦИ – РУСЕ’ 2006

```
var n, i, k, max1, max2, max3, min1, min2: longint;
begin
  max1:=-30000; max2:=max1; max3:=max1;
  min1:=30000; min2:=min1;
  readln(n);
  for i:=1 to n do begin
    readln(k);
    if k>max1 then begin
      max3:=max2; max2:=max1; max1:=k;
    end
    else if k>max2 then begin
      max3:=max2; max2:=k;
    end
    else if k>max3 then max3:=k;
    if k<min1 then begin
      min2:=min1; min1:=k;
    end
    else if k<min2 then min2:=k;
  end;
  if (max1>0) and (min1*min2>max2*max3) then writeln(max1,' ', min1,' ',min2)
  else writeln(max1,' ',max2,' ',max3);
end.
```

ТЕМА ЗА ГРУПА Е

Задача Е1. ЖИВОТИНСКА ЗАДАЧА

Кучетата и котките живеят по-кратко от нас хората, но и техният живот преминава през различни фази – бебе, дете, тийнейджър, средна възраст и старост... Интересно би било да знаем на колко човешки години се равнява възрастта на нашите домашни любимци.

Много хора смятат, че правилото 7 кучешко-котешки години са равни на 1 човешка дава точни резултати. Това, за съжаление, е погрешно и неточно изчисление... Например, повечето животни (кучета или котки) могат да имат малки на 1 година, докато е трудно си представим 7-годишно дете да бъде родител...



Всъщност, първата година се равнява на 15 човешки. Втората се равнява на 9 човешки. Всяка кучешко-котешка година от 3 до 18 се равнява на 4 години, а над 18 – на 3 години...

Така, например, ако 19-годишен домашен любимец беше човек, той щеше да е на $15+9+4+4+4+4+4+4+4+4+4+4+4+4+4+4+4+3 = 91$ човешки години.

Напишете програма **DOGGY**, която по зададена възраст на домашен любимец да изчислява човешката му възраст.

Вход: От клавиатурата се въвежда едно цяло число от 0 до 25 – реалната възраст на домашния любимец.

Изход: На екрана се извежда едно единствено цяло число – човешката възраст на домашния любимец.

ПРИМЕР:

Вход:	Изход:
--------------	---------------