

Разбор задач муниципального этапа олимпиады по информатике

1. Коробки (все классы)

Тема: разбор случаев

Сложность: простая

Нужно вычислить объемы обеих коробок, затем в зависимости от результатов сравнения нужно вывести одну из 3 строк.

Пример реализации:

```
var a,b,c,v1,v2:longint;
begin
  read(a,b,c);
  v1:=a*b*c;
  read(a,b,c);
  v2:=a*b*c;
  if v1>v2 then
    writeln('FIRST')
  else if v1<v2 then
    writeln('SECOND')
  else
    writeln('EQUAL');
end.
```

2. Алгоритм (8 класс)

Тема: реализация программы по схеме алгоритма

Сложность: простая

Пример реализации:

```
var n,k:integer;
begin
  read(n,k);
  while n>=k do
    begin
      if n>100 then k:=k+5
      else k:=k+1;
      if k<20 then n:=n-11;
      n:=n-3;
    end;
    writeln(n+k);
end.
```

2. Шифр (9-11 класс)

Тема: поиск максимума

Сложность: простая

Пример реализации:

```
var rule:array[1..1000] of integer;
    mask:array[1..10] of integer;
    n,m,i,j,s,maxi,maxs:integer;
begin
  read(n);
  for i:=1 to n do
    read(mask[i]);
  read(m);
  for i:=1 to m do
    read(rule[i]);
  maxi:=1;
  maxs:=0;
  for i:=1 to m-n+1 do
    begin
      s:=0;
      for j:=1 to n do
        if mask[j]=1 then
```

```

        s:=s+rule[i+j-1];
    if s>maxs then
    begin
        maxs:=s;
        maxi:=i;
    end;
end;
for j:=1 to n do
    if mask[j]=1 then
        write(' ',rule[j+maxi-1]);
    writeln;
end.

```

3. Пиццерия (8 класс)

Тема: поиск максимума

Сложность: простая

Пример реализации:

```

var p:array[1..100] of integer;
    n,i,j,mi,mp,tp:integer;
begin
    read(n);
    for i:=1 to n do
        read(p[i]);
    mi:=0;
    mp:=-1;
    for i:=1 to n do
    begin
        tp:=0;
        for j:=1 to n do
            if p[j]-abs(j-i)>0 then
                tp:=tp+p[j]-abs(j-i);
            if tp>mp then
            begin
                mp:=tp;
                mi:=i;
            end;
        end;
    writeln(mp,' ',mi);
end.

```

3. Пиццерия-2 (9-11 класс)

Тема: поиск в ширину

Сложность: выше среднего

Пример реализации:

```

var map,step:array[1..40,1..40] of integer;
    q:array[1..1600] of integer;
    pr,maxpr,maxi,maxj,st,q1,q2,n,m,i,j,r,c:integer;
procedure go(r,c,st:integer);
begin
    if (r<=0) or (r>n) or (c<=0) or (c>m) then exit;
    if map[r,c]<0 then exit;
    if step[r,c]>=0 then exit;
    step[r,c]:=st;
    if map[r,c]>0 then
    begin
        if map[r,c]-st+1>0 then
            pr:=pr+map[r,c]-st+1;
        exit;
    end;
    q[q2]:=r*100+c;
    inc(q2);
end;
begin

```

```

read(n,m);
for i:=1 to n do
  for j:=1 to m do
    read(map[i,j]);
maxpr:=-1;
maxi:=0;
maxj:=0;
for i:=1 to n do
  for j:=1 to m do
    if map[i,j]=0 then
      begin
        pr:=0;
        for r:=1 to n do
          for c:=1 to m do
            step[r,c]:=-1;
          step[i,j]:=0;
          q[1]:=i*100+j;
          q1:=1;
          q2:=2;
          while q1<q2 do
            begin
              r:=q[q1] div 100;
              c:=q[q1] mod 100;
              inc(q1);
              st:=step[r,c]+1;
              go(r+1,c,st);
              go(r,c+1,st);
              go(r-1,c,st);
              go(r,c-1,st);
            end;
            if pr>maxpr then
              begin
                maxpr:=pr;
                maxi:=i;
                maxj:=j;
              end;
            end;
          writeln(maxpr);
          writeln(maxi,' ',maxj);
        end.

```

4. Кругосветное путешествие (8 класс)

Тема: математика

Сложность: выше среднего

Пример реализации:

```

var diff:array[1..100000] of longint;
    n,i,x,m:longint;
begin
  read(n);
  for i:= 1 to n do
    read(diff[i]);
  for i:= 1 to n do
    begin
      read(x);
      diff[i]:= diff[i]-x;
    end;
  x:=0;
  m:=$7FFFFFFF; { 2^31-1 }
  for i:=1 to n do
    begin
      x:=x+diff[i];
      if x<m then m:=x;
    end;
  for i := 1 to n do
    begin

```

```

    if m>=0 then
        write(' ',i);
        m:=m-diff[i];
    end;
    writeln;
end.

Пример реализации с помощью моделирования путешествия (80 баллов из 100):
var fuel,dist:array[1..100000] of longint;
    i,n:integer;
function go(p:longint):boolean; { моделирование кругового путешествия }
var j,f:integer;
begin
    f:=0; { начальное количество топлива }
    for j:=1 to n do
    begin
        f:=f+fuel[p]; { заправляемся }
        f:=f-dist[p]; { едем в следующий город }
        if f<0 then { если топлива не хватило }
        begin
            go:=false; { неудача! }
            exit; { дальше можно не моделировать }
        end;
        p:=p+1; { изменяем номер города }
        if p>n then p:=1;
    end;
    go:=true; { успех! }
end;
begin
    { ввод данных }
    read(n);
    for i:=1 to n do
        read(fuel[i]);
    for i:=1 to n do
        read(dist[i]);
    for i:=1 to n do
        if go(i) then { путешествие из i-го города успешно? }
            write(' ',i); { печатаем i }
    writeln;
end.

```

4. Разбиение на палиндромы (9-11 класс)

Тема: динамическое программирование

Сложность: средняя

Пример реализации:

```

var s:string;
    dt:array[0..100] of longint;
    { dt[i] - количество разбиений на палиндромы начальной
      части строки из i символов }
    i,j,k:integer;
    palindrom:boolean;
begin
    readln(s);
    dt[0]:=1; { для пустой строки существует одно разбиение }
    for i:=1 to length(s) do
        for j:=1 to i do
            begin
                { является ли подстрока с символа j до символа i палиндромом? }
                palindrom:=true;
                for k:=0 to (i-j+1) div 2 do { проверяем до середины подстроки }
                    if s[j+k]<>s[i-k] then
                        palindrom:=false;
                if palindrom then
                    { мы знаем число способов разбить строку длиной j-1 (хранится в dt[j-1]),
                      а подстрока с j до i - это палиндром;
                      следовательно, имеем dt[j-1] новых способов деления строки длиной i }

```

```
        dt[i]:=(dt[i]+dt[j-1]) mod 1000000009; { увеличиваем количество способов }
    end;
    writeln(dt[length(s)]);
end.
```