

Решение:

Требуется построить дерево боев, т.е. схематичное изображение процесса выбора соперников. Дерево строится поэтапно. На каждом этапе бойцы вышедшие из предыдущего этапа должны провести очередной бой. Победители пар переходят на следующий этап.

Количество бойцов известно. На каждом этапе (будем использовать термин «Очередной этап») бойцов объединяем в пары. Если остался один без пары (это произойдет, если количество бойцов было нечетным), ему предоставляем бой с победителем, находящимся слева от него (это визуально видно на экране), а остальных оставляем без изменений, на этом очередной этап завершён.

Примечание:

- После каждого этапа количество бойцов уменьшается вдвое.
- Каждый этап строится, начиная с первого бойца (первого из победителей предыдущего этапа).
- Процесс продолжается до тех пор, пока не останутся 2 победителя. Они и являются финальной парой. Дерево боев построено.

В программе используется переменная n целочисленного типа для хранения количества бойцов, участвующих в соревнованиях на очередном этапе. Так как каждый из бойцов на экране обозначен точкой, то он имеет свои координаты, которые меняются по мере продвижения к финалу. Координата « y » у всех бойцов на каждом этапе совпадает и зависит только от номера этапа, а координата « x » меняется, поэтому для ее хранения заводим массив целых чисел « a ». На каждом этапе этот массив переписывается следующим образом: координата победителя есть среднее арифметическое координат участников. Абсцисса победителя первой пары записывается в $a[1]$, победителя второй – в $a[2]$ и так далее. Длина массива не меняется (она равна N) но последние элементы не используются, так как нам важны только координаты бойцов, прошедших в следующий этап.

Процедуры:

1. *Процедура ris* объединяет бойцов в пары, используя их координаты,
2. *Процедура fight* прорисовывает весь этап и является рекурсивной.

Решение на языке Turbo-Pascal

```
Program task;
uses crt,graph;
type c=array [1..100] of integer;
var
  a: c;
  n,i,driver,mode:integer;
  s:string;
procedure ris(i,y:integer);
begin
  line(a[i],y,a[i],y+10);
  line(a[i+1],y,a[i+1],y+10);
```

```

    line(a[i],y+10,a[i+1],y+10);
end;
procedure fight(var a:c; n,k:integer);
var i,y:integer;
begin
y:=45+10*(k-1);
i:=1;
while i<=n-1 do
begin
ris(i,y);
a[round((i+1)/2)]:=round((a[i]+a[i+1])/2);
i:=i+2;
end;
if i=n then
begin
line(a[n],y,a[n],y+10);
a[round((n+1)/2)]:=a[n];
end;
if n mod 2 <> 0 then
begin
n:=n div 2+1;
k:=k+1; y:=45+10*(k-1);
ris(n-1,y); a[n-1]:=round((a[n-1]+a[n])/2);
for i:=1 to n-2 do line(a[i],y,a[i],y+10);
n:=n-1;
end
else n:= n div 2;
if n>1 then fight(a,n,k+1);
end;
begin
clrscr;
writeln('количество бойцов');
readln(n);
for i:=1 to n do
a[i]:=43+30*(i-1);
driver:=detect;
initgraph(driver,mode,"");
for i:=1 to n do
begin
str(i,s);
outtextxy(a[i]-3,40,s);
end;
fight(a,n,1);
while not keypressed do;
end.

```