

Проверить, является ли массив возрастающим или убывающим или немонотонным ПО ХОДУ ДЕЛА

ПО ХОДУ ДЕЛА = если обнаружилось нарушение предыдущей монотонности (предыдущего возрастания или убывания), то прервать цикл не доходя до конца



```
void proverka_na_monotonnost2(int z[],int razmer)
{
    int i;
    int k=0;
    int s=0;
    for(i=0;i<razmer-1;i++)
    {
        if(z[i]<z[i+1])
        {
            k++;
            if(!(i>0 && z[i-1]<z[i]))
            {
                break;
            }
        }

        if(z[i]>z[i+1])
        {
            s++;
            if(!(i>0 && z[i-1]>z[i]))
            {
                break;
            }
        }
    }
    if(k==razmer-1)
    {
        printf("vozrostanie\n");
    }
    else if(s==razmer-1)
    {
        printf("ubivanie\n");
    }
    else
    {
        printf("ne monotonnost\n");
    }
}
```

```
void proverka_na_monotonnost3(int z[],int razmer)
{
    int i;
    int k=0;
    for(i=0;i<razmer-1;i++)
    {
        if(z[i]<z[i+1])
        {
            if(k==1)
            {
                k=0;
                break;
            }
            k=2;
        }
        if(z[i]>z[i+1])
        {
            if(k==2)
            {
                k=0;
                break;
            }
            k=1;
        }
    }
    if(k==2)
    {
        printf("vozrostanie\n");
    }
    else if(k==1)
    {
        printf("ubivanie\n");
    }
    else
    {
        printf("ne monotonnost\n");
    }
}
```