

```
#include <stdio.h>
```

```
#define MAX 10000
```

```
int main()
```

```
{  
    int Vpos_min[MAX] = {0}; // vector ce contine prima aparitie a numerului  
    int Vpos_max[MAX] = {0}; // vector ce contine cea mai recenta aparitie a numerului  
    int n, nr, maxlen = 0;
```

```
    scanf("%d", &n);
```

```
    for (int i = 1; i <= n; i++)
```

```
    {  
        scanf("%d", &nr);
```

```
        if (Vpos_min[nr] == 0)
```

```
            Vpos_min[nr] = i;
```

```
            Vpos_max[nr] = i;
```

```
    }
```

```
    for (int i = 0; i < MAX; i++)
```

```
        if (Vpos_min[i] && Vpos_max[i] - Vpos_min[i] + 1 > maxlen)
```

```
            maxlen = Vpos_max[i] - Vpos_min[i] + 1;
```

```
    for (int i = 0; i < MAX; i++)
```

```
        if (Vpos_min[i] && Vpos_max[i] - Vpos_min[i] + 1 == maxlen)
```

```
            printf("%d %d\n", Vpos_min[i] - 1, Vpos_max[i] - 1);
```

```
    return 0;
```

```
}
```

// Programul trece prin toate numerele aflându-le prima și ultima apariție și calculează distanța între ele pentru a afla lungimea subșirului. Apoi programul afișează pozițiile numerelor care au șiruri ce corespund cu mătineea maxlen. Programul are o complexitate timp $O(n)$ deoarece parcurge o singură dată liniar toate numerele.