/\*

PROG: baric

LANG: C++

ID: hayk.sa1

\*/

#include <stdio.h>

#include <math.h>

#include <string.h>

#define Min(a, b) ((a)<(b)?(a):(b))

int m[110];

int d[110][110];

int s[110][110], st[110], end[110];

int solve(int n, int k)

{

int &it = d[n][k];

if (it != -1)

return it;

if (k == 1)

return it = st[n];

int i;

it = 1000001;

for (i = k-1; i < n; i++)

it = Min(it, solve(i, k-1)+s[i][n]);

return it;

}

int main()

{

// freopen("baric.in", "r", stdin);

// freopen("baric.out", "w", stdout);

int n, e, i, j, k, l;

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

m[0] = 0;

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

scanf("%d", m+i);

m[n+1] = 0;

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

for (j = 1; j < i; j++)

st[i] += 2\*abs(m[i]-m[j]);

for (i = 0; i <= n; i++)

for (j = i; j <= n+1; j++)

{

s[i][j] = 0;

for (k = i+1; k < j; k++)

s[i][j] += abs(2\*m[k]-m[i]-m[j]);

}

for (i = 0; i <= n+1; i++)

for (j = i+1; j <= n; j++)

end[i] += 2\*abs(m[i]-m[j]);

memset(d, -1, sizeof(d));

for (k = 1; k <= n; k++)

{

l = 0;

  for (i = k; i <= n; i++)

if (solve(i, k)+end[i] <= e)

{

e = solve(i, k)+end[i];

l = 1;

}

if (l)

{

printf("%d %d\n", k, e);

return 0;

}

}

return 0;

}