// RMQ

int MIN(int x, int y) {return s[x] <= s[y] ? x : y;}

const int maxn = 50005;

int a[maxn], n, m;

int dp1[maxn][20]; int dp2[maxn][20];

void Init() {

mem(dp1, inf);

mem(dp2, 0);

}

void rmq() {

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

dp1[i][0] = dp2[i][0] = a[i];

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

for(int i = 1; i + (1 << j) - 1 <= n; i++) {

dp1[i][j] = min(dp1[i][j - 1], dp1[i + (1 << (j - 1))][j - 1]);

dp2[i][j] = max(dp2[i][j - 1], dp2[i + (1 << (j - 1))][j - 1]);

}

}

int query(int x, int y) {

int k = (int)(log((double)(y - x + 1)) / log(2.0));

int mmin = min(dp1[x][k], dp1[y - (1 << k) + 1][k]);

int mmax = max(dp2[x][ k], dp2[y - (1 << k) + 1][k]);

return mmax - mmin;

}

int main()

{

int u, v;

while ( scanf("%d %d", &n, &m) != EOF)

{

Init();

rep(i, 1, n)

scanf("%d", &a[i]);

rmq();

while( m-- )

{

scanf("%d %d", &u, &v);

printf("%d\n",query(u, v));

}}