<https://www.hackerrank.com/challenges/maximum-element/problem>

#include <stdio.h>

#include <string.h>

#include <math.h>

#include <stdlib.h>

int main()

{

int n,i=0;

int top = -1;

int \*st;

int \*max;

scanf("%d",&n);

st = calloc(n,sizeof(int));

max = calloc(n,sizeof(int));

while(i!=n)

{

int t,k;

scanf("%d",&t);

if(t==1)

{

scanf("%d",&k);

++top;

st[top] = k;

if((max[(top-1)]<k)||(i==0))

max[top] = k;

else

max[top] = max[(top-1)];

}

else if(t==2)

{

--top;

}

else

{

printf("%d\n",max[top]);

}

i++;

}

/\* Enter your code here. Read input from STDIN. Print output to STDOUT \*/

return 0;

}

<https://www.hackerrank.com/challenges/balanced-brackets>

#include <math.h>

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

#include <assert.h>

#include <limits.h>

#include <stdbool.h>

int mf(char a, char b)

{

if(((a =='(')&&(b == ')'))||((a=='{')&&(b == '}'))||((a=='[')&&(b == ']')))

{

//printf("1");

return 1;

}

else

return 0;

}

char\* isBalanced(char\* s)

{

int j=0;

char \*t;

int n = strlen(s);

t = calloc(strlen(s),sizeof(char));

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

{

char c = s[i];

if(!i)

{

t[j] = s[i];

++j;

continue;

}

if(mf(t[j-1],c))

{

--j;

t[j] = 'e';

}

else

{

t[j] = s[i];

++j;

}

}

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

// printf("%c",t[i]);

if(t[0] == 'e')

return "YES";

else

return "NO";

// Complete this function

}

int main() {

int t;

scanf("%i", &t);

for(int a0 = 0; a0 < t; a0++){

char\* s = (char \*)malloc(512000 \* sizeof(char));

scanf("%s", s);

int result\_size;

char\* result = isBalanced(s);

printf("%s\n", result);

}

return 0;

}