#include <stdio.h>

#include <string.h>

#include <math.h>

#include <stdlib.h>

#define STACK\_SIZE 1000000

#define MAX\_W\_SIZE ((1000000) + (1))

char\* stack[STACK\_SIZE];

int sp = -1;

int is\_empty() {

return (sp < 0);

}

int is\_full() {

return (sp > STACK\_SIZE);

}

void push(char\* cp) {

if (!is\_full()) {

stack[++sp] = cp;

}

}

char\* peek() {

char\* top = '\0';

if (!is\_empty()) {

top = stack[sp];

}

return top;

}

char\* pop() {

char\* top = peek();

if (top) {

stack[sp--] = '\0';

}

return top;

}

int get\_len(char\* warg) {

int len = 0;

while(\*warg) {

len++; warg++;

}

return len;

}

void do\_append(char\* warg) {

int len = get\_len(warg);

char\* current = peek();

if (!current) {

current = (char\*) malloc(sizeof(char) \* (len + 1));

for (int i = 0; i < len; i++) {

current[i] = warg[i];

}

current[len] = '\0';

push(current);

} else {

int j = 0;

int current\_len = get\_len(current);

char\* current\_new = (char\*)malloc(sizeof(char) \* (current\_len + len + 1));

if (current\_new) {

for (int i = 0; i < current\_len; i++) {

current\_new[i] = current[i];

}

for (int i = current\_len; i < current\_len + len; i++) {

current\_new[i] = warg[j++];

}

current\_new[current\_len + len] = '\0';

push(current\_new);

}

}

}

void do\_erase(int iarg) {

char\* current = peek();

if (current) {

int current\_len = get\_len(current);

if (current\_len >= iarg) {

char\* current\_new = (char\*)malloc(sizeof(char) \* (current\_len - iarg + 1));

if (current\_new) {

for (int i = 0; i < current\_len - iarg; i++) {

current\_new[i] = current[i];

}

current\_new[current\_len - iarg] = '\0';

push(current\_new);

}

}

}

}

void do\_get(int iarg, char\* ch) {

char\* current = peek();

if (current) {

int current\_len = get\_len(current);

if (current\_len >= iarg) {

\*ch = current[iarg - 1];

}

}

}

void do\_undo() {

pop();

}

int main() {

int Q;

int op;

int iarg;

char warg[MAX\_W\_SIZE];

scanf("%d", &Q);

for (int i = 0; i < Q; i++) {

scanf("%d", &op);

if (op == 1) {

scanf("%s", warg);

do\_append(warg);

} else if (op == 2) {

scanf("%d", &iarg);

do\_erase(iarg);

} else if (op == 3) {

scanf("%d", &iarg);

char ch = '\0';

do\_get(iarg, &ch);

fflush(stdout);

printf("%c\n", ch);

} else if (op == 4) {

do\_undo();

}

}

return 0;

}