RPNC.pdf

Ferre18

Programación de Sistemas y Concurrencia

2º Grado en Ingeniería Informática

Escuela Técnica Superior de Ingeniería Informática
Universidad de Málaga

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
  char c[5];
  while(fscanf(f, "%s", c) == 1){
   if(isOperator(c)){
    pushOperator(&stack, c[0]);
   } else {
    push(&stack, text2Int(c));
   }
  }
  int result;
  pop(&stack, &result);
  return result;
 }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
  char c[5];
  while(fscanf(f, "%s", c) == 1){
   if(isOperator(c)){
    pushOperator(&stack, c[0]);
   } else {
    push(&stack, text2Int(c));
   }
  }
  int result;
  pop(&stack, &result);
  return result;
 }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
```

```c
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
```

```c
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
  char c[5];
  while(fscanf(f, "%s", c) == 1){
   if(isOperator(c)){
    pushOperator(&stack, c[0]);
   } else {
    push(&stack, text2Int(c));
   }
  }
  int result;
  pop(&stack, &result);
  return result;
 }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```

```c
//EMPIEZA AQUI


#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include "Stack.h"

int process(char * filename);

int main(void) {
  T_Stack q;
  int ok, result;
  q = create();
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 3);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  push(&q, 4);
  push(&q, 5);
  ok = pushOperator(&q, '*');
  if (!ok) puts("* cannot operate");
  ok = pushOperator(&q, '+');
  if (!ok) puts("* cannot operate");
  push(&q, 6);
  ok = pushOperator(&q, '+');
  if (!ok) puts("+ cannot operate");
  ok = pop(&q, &result);
  if (!ok) puts("Cannot pop");
  printf("The result is %d.\n", result);
  if (isEmpty(q)) puts("Now the queue is empty.");
  else puts("Now the queue contains something.");
  result = process("source.calc");
  printf("The result from the file is %d.\n", result);
  return EXIT_SUCCESS;

}

int text2Int(char * text) {
  int value=0, i=0;
  while(isdigit(text[i]))
    value = (value*10)+(text[i++]-'0');
  return value;
}

int isOperator(char * text){
  return !isdigit(text[0]);
}

#define MAX_LENGTH 1024
int process(char * filename) {
  FILE *f = fopen(filename, "rb");
  if(f == NULL) {
    perror("Error al abrir el fichero");
    return 0;
  } else {
    T_Stack stack;
    stack = create();
    char c[5];
    while(fscanf(f, "%s", c) == 1){
      if(isOperator(c)){
        pushOperator(&stack, c[0]);
      } else {
        push(&stack, text2Int(c));
      }
    }
    int result;
    pop(&stack, &result);
    return result;
  }
}


//TERMINA AQUI
```