

1)a) O que é um tipo de dados?

É a natureza de um dado no programa, ou seja, a forma como ele será armazenado e reconhecido.

Por exemplo, o número 10, armazenado como 'char', 'int', 'float' ou 'string', ocupará diferentes espaços na memória e poderá ser usado para operações diferentes (no caso, apenas 'string' não representará um dado numérico).

b) Para que servem os tipos?

Com tipos bem definidos, o compilador (ou interpretador) pode verificar se a operação a ser feita é válida ou não, por exemplo, '3 + "3"' não é uma operação válida, já que o operador '+' não aceita o tipo 'string'.

Além disso, com tipos estáticos, como em C, o programa pode ter seu tamanho e desempenho otimizados, já que o programa não precisará armazenar memória além do que é necessário para aquele tipo e não precisará checar a validade das operações em tempo real.

2)a) O que você espera que aconteça ao compilar o programa? Por quê?

O compilador apontará um erro e não compilará, pois o operador '-' não é compatível com strings.

b) Quais são os tipos dos operandos na operação de subtração?

O 100 é um int(eiro) e o "10" uma string (cadeia de chars).

c) Transcreva a mensagem de erro de compilação e a explique.

erro.c: In function 'main':

erro.c:3:30: error: invalid operands to binary - (have 'int' and 'char *')

```
3 | printf("Soma = %d\n", 100-"10");
  |           ^
  |           |
  |           char *
```

"error: invalid operands to binary - (have 'int' and 'char *')" :

A mensagem de erro aponta que esta combinação de operandos (neste caso, 'int' e 'char *') não são válidos para esta operação (subtração).

4)a) Qual é a diferença entre tipagem estática e dinâmica?

A diferença entre tipagem estática e dinâmica está em como uma linguagem lida com seus tipos. Por exemplo, em uma linguagem com tipagem estática, o programador deve especificar os tipos das variáveis ao compilador, deste modo, qualquer conflito de tipo que possa surgir é alertado durante a compilação.

Com tipagem dinâmica, os tipos não precisam ser especificados, assim, cabe ao programa avaliar os tipos das variáveis a cada operação e convertê-los quando possível, o que geralmente leva a todas variáveis de um mesmo tipo (numérico, etc) ocuparem um mesmo espaço na memória. Além disso, as variáveis são acompanhadas por um identificador de seus tipos na memória, ocupando ainda mais espaço.

b) Dê exemplos de linguagens que usem tipagem estática e exemplos de linguagens que usem tipagem dinâmica.

- Python, JavaScript, Ruby e Lua são exemplos de linguagens com tipagem dinâmica.
- C, C++, Haskell e Java são exemplos de linguagens com tipagem estática.

c) Por quê uma linguagem com tipagem estática é tipicamente mais rápida do que uma com tipagem dinâmica? Justifique usando um exemplo.

Enquanto linguagens com tipagem estática checam se há algum erro relacionado aos tipos (operações inválidas) durante a compilação, linguagens com tipagem dinâmica precisam avaliar em tempo real se os tipos são compatíveis com a operação. Essa diferença faz com que linguagens com tipagem estática sejam mais rápidas, já que a checagem é feita apenas uma vez, antes do programa rodar.