

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Чернецкая Ирина Евгеньевна
Должность: Заведующий кафедрой
Дата подписания: 26.02.2023 06:53:09
Уникальный программный ключ:
bdf214c64d8a381b0782ea566b0dce05e3f5ea2d

МИНОБРНАУКИ РОССИИ

Юго-Западный государственный университет

УТВЕРЖДАЮ

Заведующий кафедрой

вычислительной техники

 И.Е.Чернецкая

« 01 » сентября 2022 г.

ОЦЕНОЧНЫЕ СРЕДСТВА

для текущего контроля успеваемости
и промежуточной аттестации обучающихся

по дисциплине

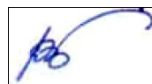
РЕКУРСИВНО-ЛОГИЧЕСКОЕ И ФУНКЦИОНАЛЬНОЕ ПРОГРАММИРОВАНИЕ

(наименование дисциплины)

09.03.02 Информационные системы и технологии

(код и наименование ОПОП ВО)

Разработчик: доцент кафедры ВТ



Ватуин Э.И.

ВОПРОСЫ ДЛЯ ЗАЧЕТА

1. Особенности языков функционального программирования (12 баллов).
2. Понятие строго функционального языка (12 баллов).
3. Представление и интерпретация функциональных программ (12 баллов).
4. Основные элементы языка Лисп: алфавит, синтаксис, семантика (12 баллов).
5. Цикл работы интерпретатора (REPL) (12 баллов).
6. Атом, список, символьное выражение, функция (12 баллов).
7. Селекторы, конструкторы и предикатные функции (12 баллов).
8. Внутреннее представление списков (12 баллов).
9. Печатное имя, значение, определение функции, список свойств (12 баллов).
10. Определение и модификация значений системных свойств (12 баллов).
11. Обобщенная функция присваивания (12 баллов).
12. Лямбда-выражение и лямбда-вызов (12 баллов).
13. Функция определения функции (12 баллов).
14. Список аргументов (12 баллов).
15. Сопоставление формальных и фактических параметров при оценивании лямбда-вызова (12 баллов).
16. Классификация управляющих структур (12 баллов).
17. Структуры следования, разветвления и повторения (12 баллов).
18. Программирование в императивном стиле средствами функционального языка (12 баллов).
19. Основы теории рекурсивных функций (12 баллов).
20. Рекурсивные функции и лямбда-исчисление А. Черча (12 баллов).
21. Средства и приемы построения рекурсивных программ (12 баллов).
22. Сравнение рекурсивного и итеративного подхода к реализации алгоритмов (12 баллов).

Для сдачи зачета студент получает 3 вопроса. Ответ на каждый вопрос оценивается по приведенным ниже критериям.

Критерии оценки:

- 12 баллов выставляется обучающемуся, если получен правильный исчерпывающий ответ на вопрос;
- от 1 до 11 баллов выставляется обучающемуся за частичный или частично правильный ответ на вопрос;
- 0 баллов выставляется обучающемуся при отсутствии ответа или неправильный ответ на вопрос.

ТЕМЫ РЕФЕРАТОВ

1. Строго функциональные языки программирования.
2. Концепция отложенных вычислений и ее использование в языках функционального программирования.
3. Объектно-ориентированные функциональные языки.
4. Особенности языка программирования Haskell.
5. Императивные и функциональные программы. Сравнение двух подходов к программированию.
6. Побочные эффекты и функциональные языки.
7. Виды рекурсии в программировании.
8. Диалекты языка LisP.
9. Стандарты языков функционального программирования (на примере Common LisP).
10. Использование функциональных языков в интеллектуальных системах.

Критерии оценки:

- 4 балла выставляется обучающемуся, если в реферате раскрыты все основные аспекты темы, выполнен достаточно глубокий анализ известных подходов и точек зрения на проблему, а также аргументированно отражено собственное мнение;
- 3 балла выставляется обучающемуся, если в реферате раскрыты все основные аспекты темы, выполнен анализ известных подходов и точек зрения на проблему;
- 2 балла выставляется обучающемуся, если в реферате раскрыты некоторые аспекты темы;
- 1 балл выставляется обучающемуся, если в реферате затронуты некоторые аспекты темы;
- 0 баллов выставляется обучающемуся при отсутствии реферата;

ВОПРОСЫ ДЛЯ СОБЕСЕДОВАНИЯ

Раздел 1. Особенности языков функционального программирования. Понятие строго функционального языка. Представление и интерпретация функциональных программ.

1. Что такое интерпретация программ и интерпретатор?
2. Чем интерпретация отличается от компиляции?
3. Что понимают по чисто функциональным языком программирования?
4. Назовите языки программирования, на которых можно записывать как функциональные, так и императивные программы.
5. Перечислите языки программирования, которые не допускают записи императивных программ.
6. Что представляет из себя программа на языке Лисп?
7. Какие структурные элементы включает функция в языке Лисп?
8. Какова роль рекурсии в программах на языке Лисп?
9. Каким образом можно отрегулировать размеры областей ввода и ответа в HomeLisp?
10. Что происходит, когда в области ввода HomeLisp записываются сразу несколько следующих подряд S-выражений?

Раздел 2. Основные элементы языка Лисп: алфавит, синтаксис, семантика. Цикл работы интерпретатора (REPL).

1. Какие группы символов входят в алфавит языка Лисп?
2. Что понимается под идентификатором?
3. Что такое литералы?
4. Какие числовые типы имеются в языке Лисп?
5. Каким образом в Лиспе представляются вещественные числа?
6. Какие аспекты правильности программ определяются синтаксисом?
7. Что такое информационная ячейка символа?
8. Какова типовая структура информационной ячейки символа?
9. Какие поля входят в информационную ячейку числа?
10. Перечислите этапы работы интерпретатора Лиспа.

Раздел 3. Атом, список, символьное выражение, точечная пара, функция.

1. Каким образом можно определить, что объект является атомом?
2. Как в Лиспе задается пустой список? Перечислите все способы.
3. Чем отличается список от символьного выражения?
4. Может ли точечная пара быть элементом списка?
5. Чем отличаются формы (SETQ x (+ 1 2)) и (SETQ x '(+ 1 2))?
6. Что задает константа NIL?

7. Для чего нужна функция QUOTE?
8. Что такое списочная ячейка?
9. Может ли точечная пара входить в список?
10. Изобразите бинарное дерево для списка ((1 2) 3 4).
11. Что делает функция CDAR?
12. Что будет результатом вызова (CDDR '(1 2 3 4))?
13. Каков результат вызова функции (CONS () NIL)?
14. Что будет результатом вызова (CADDR '(1 2 3 4))?
15. Каков результат вызова функции (CONS '(NIL) NIL)?

Раздел 6. Классификация управляющих структур. Структуры следования, разветвления и повторения. Программирование в императивном стиле средствами функционального языка.

1. Какие функции в Лиспе позволяют организовать разветвление в программе?
2. Что такое циклический процесс?
3. В чем отличие функции PROG от PROGN?
4. Сколько форм можно записывать внутри предложения PROGN?
5. Что такое неявный PROG?
6. Сколько альтернативных ветвей можно реализовать с помощью функции COND?
7. Опишите, какие значения возвращает функция IF, если у нее нет третьего аргумента.
8. Какие функции в HomeLisp позволяют реализовать циклический процесс?
9. Где в теле функции LOOP должно размещаться условие прерывания цикла?
10. Сколько условий прерывания цикла можно записать в теле функции LOOP?
11. Для чего нужна функция RETURN?
12. Какие функции для организации циклов, кроме LOOP, имеются в среде HomeLisp?
13. Опишите назначение функции GO. Почему стремятся исключить необоснованное использование этой функции?

Критерии оценки:

- 4 балла выставляется обучающемуся, если получен исчерпывающий ответ на вопрос, включающий теоретические аспекты, обоснование, иллюстративный пример;
- 3 балла выставляется обучающемуся, если получен частичный ответ на вопрос, включающий теоретические аспекты и иллюстративный пример, обоснование отсутствует или недостаточное;

- 2 балла выставляется обучающемуся, если получен частичный ответ на вопрос, включающий теоретические аспекты или иллюстративный пример, обоснование отсутствует или неверное;
- 1 балл выставляется обучающемуся, если получен поверхностный ответ на вопрос, включающий некоторые теоретические аспекты или элементы иллюстрации, обоснование отсутствует или неверное;
- 0 баллов выставляется обучающемуся при отсутствии ответа.

ТЕСТОВЫЕ ЗАДАНИЯ

(ограниченная выборка из базы тестовых заданий)

Уровень сложности 1

Какой (какие) из перечисленных языков не является функциональным: Miranda, Haskell-98, Clean, Basic, Lisp?

Clean
Basic
Miranda
Miranda и Clean
Miranda и Basic

Какой язык называют чисто функциональным?

тот, на котором можно писать в том числе и функциональные программы
тот, на котором можно писать в том числе и императивные программы
тот, на котором можно программировать вызовы функций
тот, на котором можно писать только функциональные программы
тот, на котором можно программировать вызовы рекурсивных функций

Что понимается под аргументом функции?

это некоторое значение из области определения функции, которое используется в теле функции
это некоторое обозначение, которое используется при записи тела функции
это некоторое значение из области определения функции, которое подставляется в функцию при ее вызове
это некоторое значение из области определения функции, которое возвращается функцией в качестве результата
это некоторое значение из области определения функции, которое позволяет параметризовать определение функции

Какое из сформулированных ниже утверждений неверно?

Определение функции всегда должно иметь имя
Программа на языке ЛИСП является функцией
Определение функции включает список формальных параметров
Определение функции начинается открывающей круглой скобкой
Язык Лисп позволяет выполнять присваивания значений

Что будет результатом интерпретации скобочной формы (car '((1 2) (3 4)))?

1
1 2
(1 2)
T
NIL

Уровень сложности 2

Какое условие проверяется в следующей форме: $(\text{NOT} (\text{EQ } N \ 2))$?

N эквивалентно 2
N не эквивалентно 2
N равно 2
N не равно 2
это ошибочная конструкция

Какую проверку осуществляет следующая форма:
 $(\text{AND} (\text{EQ} (\text{MOD } N \ 2) \ 0) (\text{NOT} (\text{EQ } N \ 2)))$?

N четно или N не равно 2
N четно и N эквивалентно 2
N четно и N не равно 2
N нечетно и N не равно 2
конструкция записана с ошибкой

Какое действие задает следующая форма: $(\text{car}(\text{cdr}(\text{car } \text{lst})))$?

выдает второй элемент первого подсписка списка lst
выдает первый элемент первого подсписка списка lst
выдает второй элемент второго подсписка списка lst
выдает второй элемент второго подсписка списка lst
выдает второй элемент последнего подсписка списка lst

Какое действие реализует следующая форма:
 $(\text{COND} ((\text{EQ } \text{FLAG} \ 0) \ \text{NIL}) (\text{T } \text{T}))$?

если $\text{FLAG} \neq 0$, то выдать NIL, иначе выдать T
если $\text{FLAG} = 0$, то выдать NIL, затем выдать T
если $\text{FLAG} = 0$, то выдать NIL, иначе выдать T
если $\text{FLAG} = 0$, то выдать NIL, иначе выдать (T T)
конструкция ошибочна

Что понимается под отображающим функционалом?

это функция, возвращаемым значением которой является функция
это функция, все параметры которой являются функциями
это функция, применяющая функцию своего первого аргумента к остальным аргументам
это функция, применяющая некоторую функцию-аргумент к элементам списков-аргументов
это функция, применяющая функцию своего последнего аргумента к первым аргументам

Уровень сложности 3

Какое преобразование списка выполняет следующая программа:

```
(defun rmf(el spis)
  (cond
    ((null spis) nil)
    ((eq el (car spis)) (cdr spis))
    ( T (cons (car spis) ( rmf el (cdr spis) ) ) )
  ))
```

вычисляет количество вхождений элемента el в список spis
удаляет первое вхождение элемента el из списка spis
удаляет последнее вхождение элемента el из списка spis
удаляет все вхождения элемента el из списка spis
заменяет первое вхождение элемента el в список spis нулевым значением

Чему будет равно значение переменной FLAG после выполнения следующего цикла, если LST = (1 2 3 4 5)?

```
(SETQ FLAG 0)
(LOOP
  ( (NULL LST) FLAG )
  ( (EQ (CAR LST) 2) (SETQ FLAG 1) )
  ( SETQ LST (CDR LST) )
)
```

0
2
1
T
NIL

Какое значение будет сформировано в результате выполнения функции, если X=7 и LST = (1 3 5 7 8)?

```
(DEFUN POSITION (X LST)
  (COND ( (NULL LST) 0 )
        ( (EQ X (CAR LST)) 1 )
        ( (MEMBER X LST)
          (+ 1 (POSITION X (CDR LST))) )
        ( T 0 )
  ))
```

0
4
3
никакое
NIL

Уровень сложности 4

Какое действие выполняет следующая функция?

```
(DEFUN F1 (X)
  (COND
    ( (ATOM X) X)
    ((AND (LISTP X) (EQ (CAR X) 'QUOTE))
     (F1 (CADR X) ))
    ( T (CONS (F1 ( CAR X )) (F1 ( CDR X ) ) ) )
  )
```

заменяет все функции QUOTE их первыми аргументами на первом уровне вложенности
заменяет все функции QUOTE их вторыми аргументами на всех уровнях вложенности
заменяет все функции QUOTE их последними аргументами на всех уровнях вложенности
заменяет все функции QUOTE их первыми аргументами на всех уровнях вложенности
заменяет все функции QUOTE их первыми аргументами на последнем уровне вложенности

Какой результат выдаст следующая функция F1, если ее параметры X = (1 2 3 4), K = 2?

```
(DEFUN F1 (X K)
  (IF (NULL X) NIL
      (CONS (+ K (CAR X)) (F1 (CDR X) (+ K 1) ) )
  )
```

(3 4 5 6)
(3 5 7 9)

(2 3 4 5)

(1 2 3 4)

(6 6 6 6)

Дана функция, зеркально отображающая список вместе с подписками всех уровней вложенности. Какая в ней допущена ошибка?

```
(DEFUN REVERSE(lst)
```

```
(COND ((NULL lst) NIL)
```

```
      ((ATOM (CAR LST)) (APPEND (REVERSE (CDR lst)) (LIST(CAR LST)))) )
```

```
      (T (APPEND (REVERSE (CDR lst)) (REVERSE (CAR LST)))) )
```

```
)
```

должно быть (T (APPEND (REVERSE (CAR lst)) (LIST (REVERSE (CAR LST)))))

должно быть (T (APPEND (REVERSE (CAR lst)) (REVERSE (CAR LST)))))

должно быть (T (APPEND (REVERSE (CAR lst)) (LIST (REVERSE (CDR LST)))))

должно быть (T (APPEND (REVERSE (CAR lst)) (REVERSE (CDR LST)))))

должно быть (T (APPEND (REVERSE (CDR lst)) (LIST (REVERSE (CAR LST)))))

Уровень сложности 5

Дана программа, которая выводит все числа от 1 до N, делящиеся без остатка на 3, 5 и 7. Какая в этой программе есть ошибка?

```
(DEFUN DA1 (N)
```

```
  (SETQ L 0)
```

```
  (SETQ I 1)
```

```
  ( LOOP
```

```
    ( (> I N) ((PRINT "Всего чисел") L) )
```

```
    (COND ( (OR (ZEROP (CDR (DIVIDE I 3)))
```

```
              (ZEROP (CDR (DIVIDE I 5)))
```

```
              (ZEROP (CDR (DIVIDE I 7))))
```

```
      ( (SETQ L (+ L 1)) (PRINT I) ) )
```

```
    (SETQ I (+ I 1))
```

```
  )
```

```
)
```

в COND-форме вместо OR нужно записать AND

в COND-форме вместо OR нужно записать XOR

вместо CDR везде нужно записать CAR

вместо (> I N) нужно записать (<= I N)

вместо (> I N) нужно записать (> N I)

Какую задачу решает следующая функция?

```
(DEFUN F (K)
  (COND
    ((EQ K 1) 1)
    ((EQ K 2) 1)
    (T ((SETQ I 3) (SETQ F1 1)
        (SETQ F2 1) (SETQ F3 2)
        (LOOP
          ((> I K) F3)
          (SETQ F3 (+ F1 F2))
          (SETQ F1 F2) (SETQ F2 F3)
          (SETQ I (+ I 1))))))
  )
)
```

- | |
|---|
| возвращает K-й элемент последовательности простых чисел |
| возвращает K-й элемент последовательности факториалов чисел |
| возвращает K-й элемент последовательности избыточных чисел |
| возвращает K-й элемент последовательности чисел Фибоначчи |
| возвращает K-й элемент последовательности совершенных чисел |

Дана функция, проверяющая является ли число N простым. Какую форму нужно вставить в эту функцию вместо символа многоточие?

```
(DEFUN SIMPLE (N)
  (SETQ FLAG 1)
  (COND ((AND (EQ (MOD N 2) 0) (NOT (EQ N 2))) NIL)
    (T ((SETQ I 3)
        (LOOP
          ((< (CAR (DIVIDE N 2)) I) )
          ((EQ (CDR (DIVIDE N I)) 0)
            ... )
          (SETQ I (+ I 2))
        )
        (COND ((EQ FLAG 0) NIL)
              (T T )))))
  )
)
```

- | |
|--------------------|
| (SETQ I (+ I 1)) |
| (EQ FLAG 0) |
| (EQ FLAG 1) |
| (SETQ FLAG 1) |
| (SETQ FLAG 0) |

Ответы на вопросы оцениваются по приведенным ниже критериям.

Критерии оценки:

- вопрос уровня сложности 1 – 4 балла;
- вопрос уровня сложности 2 – 6 баллов;
- вопрос уровня сложности 3 – 8 баллов;
- вопрос уровня сложности 4 – 12 баллов;
- вопрос уровня сложности 5 – 20 баллов.

База тестов содержит не менее 100 вопросов (тестовых заданий), подобных представленным выше. Уровень сложности вопроса определяет количество баллов, которое студент получает в случае правильного ответа. Студенту предлагается ответить на 10 вопросов, выбранных случайным образом (по 2 вопроса каждого уровня сложности). Максимальное число баллов, которое студент может набрать в ходе тестирования, равно 100. Набранное число баллов пропорционально пересчитывается по шкале баллов (от 0 до 36), получаемых на зачете.