

## Курсовая работа. Методические указания по выполнению

---

Курсовая работа должна выполняться после изучения всего теоретического материала и выполнения всех лабораторных работ, поскольку основывается на программах, написанных в ходе выполнения лабораторных работ.

Каждый студент выполняет несколько задач из приведенного ниже списка в соответствии со своим номером варианта. На проверку преподавателю необходимо высылать:

1. Исходный текст программы.
2. Откомпилированный модуль – ехе-файл.
3. Отчет по выполненной работе.

Отчет должен содержать:

1. титульный лист (обязательно указывать номер варианта);
2. текст задания;
3. описание основных алгоритмов, использованных в программе;
4. описание основных переменных;
5. описание основных блоков (подпрограмм) программы;
6. текст программы;
7. результаты (копия экрана).

### Задание:

Необходимо написать программу, реализующую параллельную работу нескольких процессов. Каждый процесс может состоять из одного или нескольких потоков. Любой из потоков, работающих в составе этих процессов, может быть приостановлен и вновь запущен некоторой определенной клавишей (можно использовать буквенные или цифровые клавиши). Нажатия клавиш обрабатывать с помощью прерывания от клавиатуры (по материалам лаб. работы №1).

Окончание работы программы должно происходить при приостановке всех потоков их ключевыми клавишами либо при нажатии клавиши ESC. При окончании работы необходимо выполнить корректное завершение, т.е. «дочитать» всю информацию из буфера каждого процесса (при его наличии), закрыть все открытые файлы и т.п. – по материалам лаб. работы №3.

Те потоки, которые выводят информацию на экран, должны использовать для этого каждый свое отдельное окно, обрамленное рамкой – экран визуально делится на несколько окон, в каждом из которых отображается один процесс. Дополнительное информационное окно должно содержать описание управляющих клавиш и краткое сообщение о тех действиях, которые выполняются в программе: номер потока, его статус в текущий момент времени (активен, приостановлен, ожидает), другую информацию (указание величины выделенного потоку кванта времени, процент заполненности буфера и т.п.) в зависимости от конкретного потока.

Программа должна работать устойчиво, без «зависаний» и непредвиденных зрительных эффектов. В случае необходимости указания параметров при запуске программы необходимо предусмотреть и обработать все возможные ошибки (с выводом диагностики на экран).

Задачи 1, 3, 6 предполагают наличие управляющего блока, который, используя прерывания таймера, случайным образом определяет очередной активный поток и выделяет ему кванты времени (возможно, в рамках большего кванта, выделенного всему процессу). Кванты времени – как для процесса в целом, так и для его потоков – изначально задавать некоторой фиксированной величины, но предусмотреть возможность её изменения во время работы путем нажатия некоторых ключевых клавиш (для каждого из потоков предусмотреть свою клавишу) – т.е. можно, например, увеличить или уменьшить квант только потока-производителя. При этом допустимы разные варианты реализации – «общий» квант может либо меняться, либо оставаться постоянным (тогда при ускорении одного потока другой автоматически замедлится, т.к. величина его кванта уменьшится).

Потоки этого класса задач могут иметь три статуса: «активен», «ожидает» или «приостановлен». В процессе работы может возникнуть, например, следующая ситуация. Поток-потребитель приостановлен своей ключевой клавишей, следовательно,

буфер не освобождается. Поток-производитель активен, он заполнит буфер информацией и перейдёт в состояние ожидания. Из этого состояния он сможет выйти только после того, как будет возобновлена (нажатием клавиши) работа потока-потребителя, который освободит место в буфере для помещения новой информации. В случае приостановки производителя возникнет аналогичная ситуация, только с пустым буфером. Приостановка/возобновление потока возможны в любом его состоянии – как в активном, так и в состоянии ожидания.

Задачи 2, 4, 5, 7 должны выполняться равномерно, независимо от степени загрузки системы. Для этого каждой из них необходимо получать управление через фиксированное количество «тиков» системного таймера, во время которого они выполняют какое-то свое элементарное действие («бегущая строка» или «летающий объект» сместится на одну позицию, сменится нота в музыке...). При такой реализации скорость каждого потока будет определяться количеством «тиков» таймера между его запусками. Для уменьшения скорости такого потока достаточно после нажатия ключевой клавиши предоставлять ему управление реже, через большее число «тиков», соответственно для ускорения такого потока – опять же после нажатия ключевой клавиши – ему предоставляется управление чаще, в пределе – на каждом «тике». Так, «бегущая строка» должна двигаться равномерно с постоянной скоростью (если она не приостановлена ключевой клавишей) независимо от количества активных процессов в системе, музыка – тоже играть равномерно... Потоки этого класса задач могут иметь два статуса: «активен» и «приостановлен».

#### **ЗАДАЧИ:**

1. Два потока: первый читает информацию из файла (например, стихи или текст программы) в буфер, второй эту информацию из буфера выдаёт на экран. При заполнении окна вывода до конца его содержимое не должно обновляться полностью – вывод новой информации должен осуществляться в последнюю строку, а все остальные строки смещаться вверх (по материалам лаб. работы №3). Имя читаемого файла задавать как параметр командной строки. После окончания файла он начинает считываться заново.
2. Музыкальное сопровождение – как минимум несколько нот различной длительности, образующие мелодию. Во включённом состоянии мелодия должна играть в фоновом режиме (по материалам лаб. работы №2).
3. Два потока: один выполняет поиск всех последовательных простых чисел и заносит их в буфер, второй осуществляет их вывод из буфера на экран (по материалам лаб. работы №3).
4. Вывод заголовка работы (фамилия и имя автора и название работы) постоянно меняющегося цвета. Место вывода (верхняя или нижняя строка экрана) задавать в качестве параметра командной строки.
5. Вывод заголовка работы (фамилия и имя автора и название работы) в виде «бегущей строки». Место вывода (верхняя или нижняя строка экрана) задавать параметром командной строки.
6. Два потока: один генерирует случайные числа и выбирает в буфер те, которые являются полным квадратом, второй выводит их из буфера на экран (по материалам лаб. раб. №3).
7. Движущийся объект (в простейшем случае, например, летающий и отражающийся от границ окна шарик). Движение может осуществляться свободно или при управлении с клавиатуры.
8. В углу экрана с заданными в качестве параметров координатами поместить часы, показания которых должны соответствовать системному времени. Обновлять показания часов каждую секунду (по материалам лаб. работы №2).

#### **ВАРИАНТЫ заданий:**

<i>номер варианта</i>	<i>номера задач</i>	<i>номер варианта</i>	<i>номера задач</i>	<i>номер варианта</i>	<i>номера задач</i>
<b>1</b>	1, 2, 3, 5	<b>9</b>	1, 3, 5, 8	<b>17</b>	2, 3, 5, 7
<b>2</b>	1, 2, 5, 7	<b>10</b>	1, 5, 6, 8	<b>18</b>	2, 3, 4, 6
<b>3</b>	1, 2, 5, 6	<b>11</b>	1, 5, 7, 8	<b>19</b>	3, 4, 6, 7
<b>4</b>	1, 2, 5, 8	<b>12</b>	2, 3, 4, 7	<b>20</b>	3, 5, 7, 8
<b>5</b>	1, 2, 4, 7	<b>13</b>	2, 4, 6, 7	<b>21</b>	3, 5, 6, 7
<b>6</b>	1, 4, 7, 8	<b>14</b>	2, 5, 6, 7	<b>22</b>	3, 5, 6, 8
<b>7</b>	1, 3, 4, 7	<b>15</b>	2, 3, 5, 8	<b>23</b>	5, 6, 7, 8
<b>8</b>	1, 4, 6, 7	<b>16</b>	2, 5, 6, 8	<b>24</b>	4, 6, 7, 8



