

生产者与消费者问题算法实现

一、实验目的和要求

全面理解生产者与消费者问题模型，掌握解决该问题的算法思想，正确使用同步机制。

二、实验内容、方法和步骤

问题描述：一组生产者向一组消费者提供消息，它们共享一个有界缓冲池，生产者向其中投放消息，消费者从中取得消息。假定这些生产者和消费者互相等效，只要缓冲池未滿，生产者可将消息送入缓冲池；只要缓冲池未空，消费者可从缓冲池取走一个消息。

功能要求：根据进程同步机制，编写一个解决上述问题的可视化程序，可显示缓冲池状态、放数据、取数据等过程。

具体参数：有 4 个生产者进程，分别为 P1、P2、P3 和 P4；

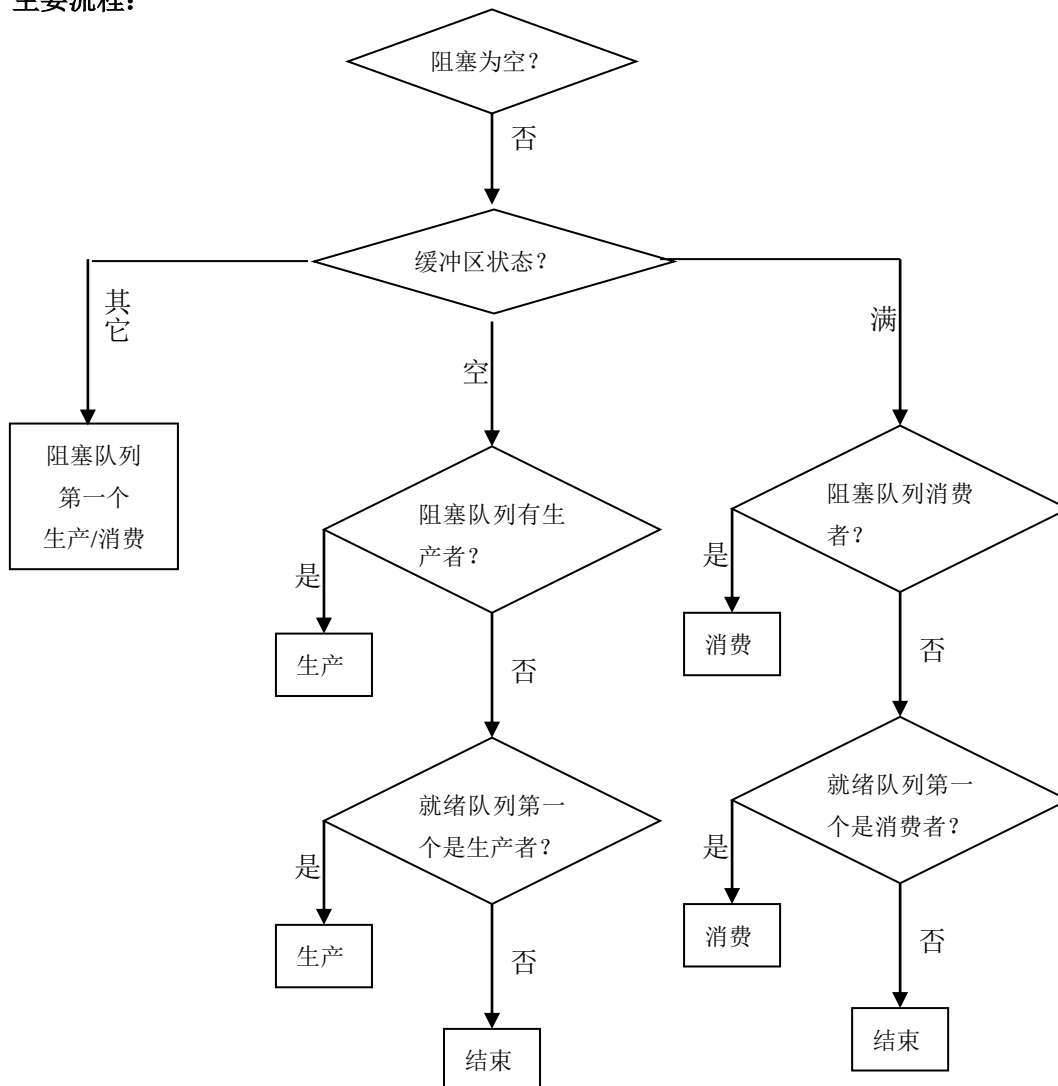
有 4 个消费者进程，分别是 C1、C2、C3 和 C4；

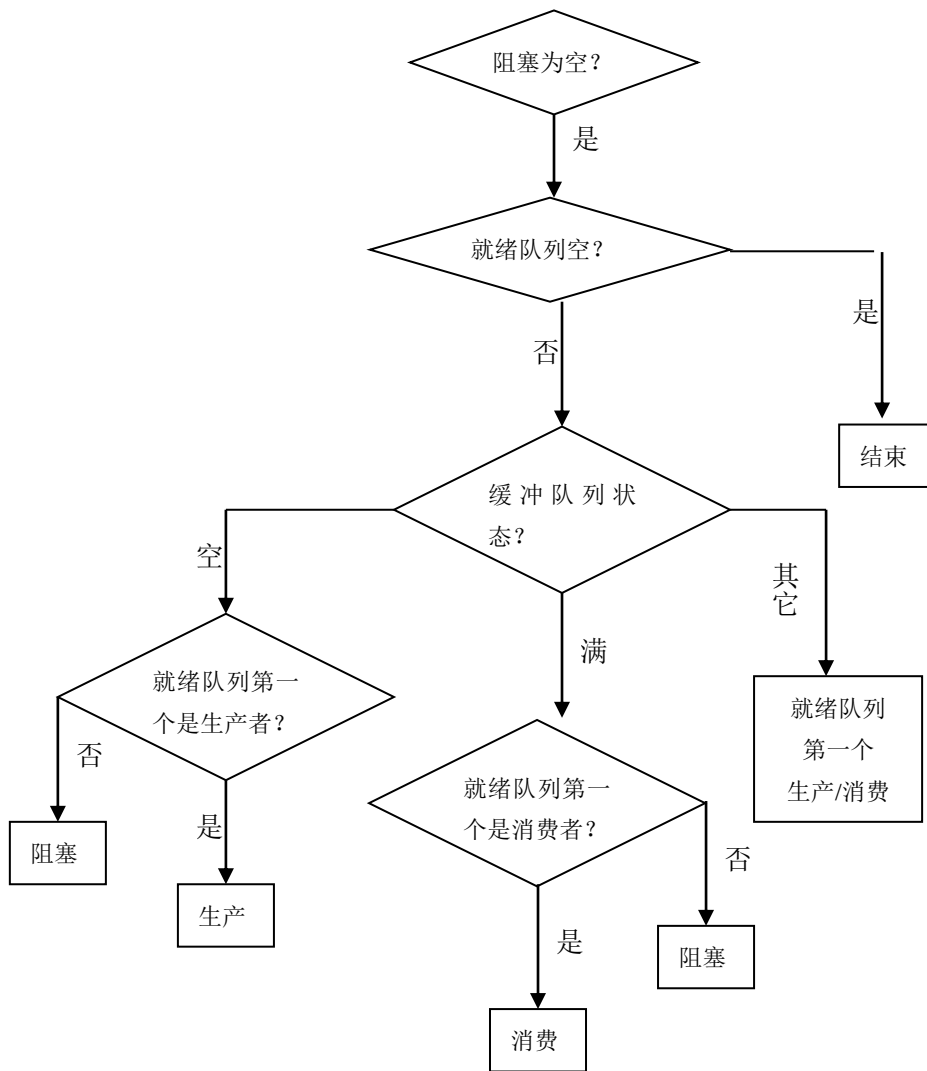
缓冲区单元个数 $N=15$ ；

不同的生产进程可生产不同的产品，自己可任意定义，在本程序中是直接将相应生产者代号放入缓冲区，消费时将某消费者消费某生产者的信息显示出来

三、设计流程图

主要流程：





三、数据定义

MyArray=array[0..14] of string;

hcdl,zsdl,jxdl:MyArray; //定义缓冲队列、阻塞队列和就绪队列，大小为 15

function sta(dl:MyArray):string; //函数：判断指定队列空、满或有值

function searchPC(pc:string;dl:MyArray):integer; // 函数：从指定列找出生产者或消费者进程，返回该元素在阻塞队列的序号

function dlval(dl:MyArray;pos:integer):string; //函数，得到指定队列指定元素的内容，并判断其类型

procedure PC(var dl:MyArray;pos:integer); //过程，根据参数所指定的队列和元素进行生产或消费

procedure ZS(var dl:MyArray;pos:integer); // 过程，将指定队列中指定元素送入阻塞队列

四、源程序

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, ExtCtrls, ComCtrls, Buttons, StrUtils;

type
  TForm1 = class(TForm)
    GroupBox1: TGroupBox;
    GroupBox2: TGroupBox;
    GroupBox3: TGroupBox;
    ListBox1: TListBox;
    ListBox2: TListBox;
    ListBox3: TListBox;
    GroupBox4: TGroupBox;
    GroupBox5: TGroupBox;
    Button1: TButton;
    Edit1: TEdit;
    Label2: TLabel;
    memo1: TMemo;
    BitBtn1: TBitBtn;
    BitBtn2: TBitBtn;
    BitBtn3: TBitBtn;
    Label1: TLabel;
    memo2: TMemo;
    Label3: TLabel;
    procedure Button2Click(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure Button1Click(Sender: TObject);
    procedure BitBtn2Click(Sender: TObject);
    procedure BitBtn1Click(Sender: TObject);
    procedure BitBtn3Click(Sender: TObject);

  private
    { Private declarations }
  public
    { Public declarations }
  end;

type
  MyArray=array[0..14] of string;
var
  Form1: TForm1;
  memo1: TMemo;
  memo2: TMemo;
  hcdl,zsdl,jxdl:MyArray;
  steps:integer=0;
  function sta(dl:MyArray):string;
  function searchPC(pc:string;dl:MyArray):integer;
  function dlval(dl:MyArray;pos:integer):string;
  procedure PC(var dl:MyArray;pos:integer);
  procedure ZS(var dl:MyArray;pos:integer);
implementation

{$R *.dfm}

procedure TForm1.Button2Click(Sender: TObject);
begin
  self.Close;
end;

procedure TForm1.FormActivate(Sender: TObject);
var
  i:integer;
begin
  listbox1.Items.Clear;
```

```

        listbox2.Items.Clear;
        listbox3.Items.Clear;
        memo1.Clear;
        memo2.Clear;
        edit1.Clear;
        bitbtn3.Enabled:=false;
        steps:=0;
    for i:=0 to 14 do
        begin
            hcdl[i]:= "";
            zsdl[i]:= "";
            jxdl[i]:= "";
            listbox1.Items.Add("");
            listbox2.Items.Add("");
            listbox3.Items.Add("");
        end;

end;

procedure TForm1.Button1Click(Sender: TObject);
var
    num,rnd,i:integer;

begin
    if edit1.text="" then
        begin
            exit;
        end;
    num:=strtoint(edit1.text);
    if (num>15) or (num<=0) then
        begin
            showMessage ('请输入 1-15 的数字！ ');
            edit1.Clear;
            edit1.SetFocus;
        end
    else
        begin
            for i:=0 to num-1 do
                begin
                    Randomize;
                    rnd:=random(8)+1;
                    if (rnd>=1)and (rnd<=4) then
                        jxdl[i]:='P'+inttostr(rnd)
                    else
                        jxdl[i]:='C'+inttostr(rnd-4);
                end;
            listbox3.Items.Clear;
            for i:=low(jxdl) to num-1 do
                begin
                    listbox3.Items.Add(jxdl[i]);
                end;
            steps:=steps+1;
            memo2.Lines.Add(inttostr(steps)+'： 随机生成了'+inttostr(num)+'个进程！ ');

            for i:=num to high(jxdl) do
                begin
                    listbox3.Items.Add("");
                end;
            end;
            bitbtn3.Enabled:=true;
        end;

end;

procedure TForm1.BitBtn2Click(Sender: TObject);
begin
    self.Close;
end;

procedure TForm1.BitBtn1Click(Sender: TObject);
begin

```

```

self.FormActivate(Sender);
end;

procedure TForm1.BitBtn3Click(Sender: TObject);
var
    zsdlsa,hcdlsa,jxdlsa:string;
    jxtp:string;
begin
    zsdlsa:=sta(zsdl);//判断阻塞队列状态，空与非空
    hcdlsa:=sta(hcdl);//判断缓冲池状态
    jxdlsa:=sta(jxdl);//就绪队列状态
    jxtp:=dlval(jxdl,0);//判断就绪队列第一个进程的类型
    if zsdlsa<>'empty' then
        begin
            //阻塞不为空，就从阻塞队列拿
            {steps:=steps+1;
            memo2.Lines.Add(inttostr(steps)+'： 阻塞队列不为空');
            }
            if hcdlsa='empty' then //缓冲为空，需要生产者
                begin
                    // showmessage('在阻塞中找生产者--生产，找不到就从就绪队列拿第一个元素
生产/消费，若就绪为空，结束');
                    if searchPC('P',zsdl)<>-1 then
                        begin
                            steps:=steps+1;
                            memo2.Lines.Add(inttostr(steps)+'： 在阻塞队列中找到生产者');
                            //生产
                            PC(zsdl,searchPC('P',zsdl));
                            end
                        else if jxdlsa<>'empty' then //就绪队列不为空
                            begin
                                steps:=steps+1;
                                memo2.Lines.Add(inttostr(steps)+'： 阻塞队列中无生产者'); //查看就绪队列第一
个进程);
                                if jxtp='P' then
                                    begin
                                        steps:=steps+1;
                                        memo2.Lines.Add(inttostr(steps)+'： 就绪队列第一个进程是生产， 生产');
                                        PC(jxdl,0);
                                        end
                                    else if jxtp='C' then
                                        begin
                                            steps:=steps+1;
                                            memo2.Lines.Add(inttostr(steps)+'： 就绪队列第一个进程是消费， 阻塞');
                                            ZS(jxdl,0);
                                            end
                                        else
                                            begin
                                                steps:=steps+1;
                                                memo2.Lines.Add(inttostr(steps)+'： 程序出错-就绪队列中出现的非法进
程！ ');
                                                exit;
                                                end;
                                            end
                                        else
                                            //就绪队列为空
                                            begin
                                                steps:=steps+1;
                                                memo2.Lines.Add(inttostr(steps)+'： 缓冲为空，需要生产者，程序结束');
                                                exit;
                                                end;
                                            end
                                        else if hcdlsa='full' then //缓冲满，需要消费者
                                            begin
                                                steps:=steps+1;
                                                // showmessage('在阻塞中找消费者--消费，找不到就从就绪队列拿第一个元素
生产/消费，若就绪为空，结束');
                                                if searchPC('C',zsdl)<>-1 then
                                                    begin
                                                        steps:=steps+1;
                                                        memo2.Lines.Add(inttostr(steps)+'： 在阻塞队列中找到消费者');
                                                        //消费

```

```

        PC(zsdl,searchPC('C',zsdl));
    end
else
    if jxdlsta<>'empty' then    //就绪队列不为空
    begin
        steps:=steps+1;
        memo2.Lines.Add(inttostr(steps)+'： 阻塞队列中无消费者');//查看就绪队列第一
个进程');
        if jxtop='P' then
        begin
            steps:=steps+1;
            memo2.Lines.Add(inttostr(steps)+'： 就绪队列第一个进程是生产， 阻塞');
            ZS(jxdl,0);
        end
        else if jxtop='C' then
        begin
            steps:=steps+1;
            memo2.Lines.Add(inttostr(steps)+'： 就绪队列第一个进程是消费， 消费');
            PC(jxdl,0);
        end
        else
        begin
            steps:=steps+1;
            memo2.Lines.Add(inttostr(steps)+'： 程序出错-就绪队列中出现的非法进
程！ ');
            exit;
            end;
        end
    else
        //就绪队列为空
        begin
            steps:=steps+1;
            memo2.Lines.Add(inttostr(steps)+'： 缓冲为空， 需要消费者， 程序结束');
            exit;
            end;
    end
else
    //缓冲区有元素也有空位， 则 可生产可消费
    begin
        steps:=steps+1;
        memo2.Lines.Add(inttostr(steps)+'： 运行阻塞队列第一个进程');
        PC(zsdl,0);
    end
end
else if zsdlsta='empty' then
    begin
        //阻塞为空 ， 就从就绪队列拿一个
        { steps:=steps+1;
        memo2.Lines.Add(inttostr(steps)+'： 阻塞队列为空'); }
    if jxdlsta<>'empty' then    //就绪队列不为空,可以从里面拿
    begin
        if hcdlsta='empty' then
        begin
            if jxtop='P' then
            begin
                steps:=steps+1;
                memo2.Lines.Add(inttostr(steps)+'： 就绪队列第一个进程是生产， 生产');
                PC(jxdl,0);
            end
            else
            begin
                steps:=steps+1;
                memo2.Lines.Add(inttostr(steps)+'： 就绪队列第一个进程是消费， 阻塞');
                ZS(jxdl,0);
            end;
        end
        else if hcdlsta='full' then
        begin
            if jxtop='C' then
            begin
                steps:=steps+1;

```

```

        memo2.Lines.Add(inttostr(steps)+'：就绪队列第一个进程是消费，消费');
        pc(jxdl,0);
    end
else
    begin
        steps:=steps+1;
        memo2.Lines.Add(inttostr(steps)+'：就绪队列第一个进程是生产，阻塞');
        ZS(jxdl,0);
    end;
end

else
    //缓冲区有元素也有空位，则 可生产可消费
    begin
        if jxtop='P' then
            begin
                steps:=steps+1;
                memo2.Lines.Add(inttostr(steps)+'：就绪队列第一个进程是生产，
生产');
                PC(jxdl,0);
            end
        else if jxtop='C' then
            begin
                steps:=steps+1;
                memo2.Lines.Add(inttostr(steps)+'：就绪队列第一个进程是消费，
消费');
                PC(jxdl,0);
            end;
        end;
    end
else
    //就绪队列为空,程序结束
    begin
        steps:=steps+1;
        memo2.Lines.Add(inttostr(steps)+'：无任何进程可运行，程序结束！');
        exit;
    end
end;
end;

```

////////////////////////////////////

//函数：判断指定队列空、满或有值

function sta(dl:MyArray):string;

var

notfull,notempty:integer;

i:integer;

begin

notfull:=0;

notempty:=0;

for i:=0 to 14 do

begin

if dl[i]=" " then

notfull:=1

else

notempty:=1;

end;

if (notfull=1) and (notempty=0) then

result:='empty'

else if (notfull=0) and (notempty=1) then

result:='full'

else if (notfull=1) and (notempty=1) then

result:='ok';

end;

// 函数：从指定列找出生产者或消费者进程，返回该元素在阻塞队列的序号

```

function searchPC(pc:string;dl:MyArray):integer;
var
i:integer;
begin
    result:=-1;
    for i:=0 to 14 do
    begin
        if leftstr(dl[i],1)=pc then
        begin
            result:=i;
            exit;
        end;
    end;
end;

//  SHOWMESSAGE('阻塞队列无需要的进程');
end;

////////函数，得到指定队列指定元素的内容，并判断其类型
function dlval(dl:MyArray;pos:integer):string;
begin
    if leftstr(dl[pos],1)='P' then
        result:='P'
    else if leftstr(dl[pos],1)='C' then
        result:='C';
    end;
end;
////////过程，根据参数所指定的队列和元素进行生产或消费
procedure PC(var dl:MyArray;pos:integer);
var
i,t:integer;
begin
    if leftstr(dl[pos],1)='P' then
    begin
        steps:=steps+1;

        form1.memo2.Lines.Add(inttostr(steps)+'： '+dl[pos]+'生产一个，缓冲加一，队列减一');

        for i:=0 to 14 do
        begin
            if hcdl[i]=" then
            begin
                hcdl[i]:=dl[pos];
                steps:=steps+1;
                form1.memo2.Lines.Add(inttostr(steps)+'： '+dl[pos]+'生产： '+hcdl[i]);
                form1.listbox1.Items.Clear;
                for t:=0 to 14 do //在列表显示出来
                begin
                    form1.listbox1.Items.Add(hcdl[t]);
                end;
                for t:=pos to 13 do
                begin
                    dl[t]:=dl[t+1]; //队列中撤消进程
                    dl[t+1]:=";
                end;
                // showmessage(dl[pos]);
                form1.listbox3.Items.Clear;
                for t:=0 to 14 do //在列表显示出来
                begin
                    form1.listbox3.Items.Add(dl[t]);
                end;
            end;
            exit; //退出
        end;

    end;

end;

end
else if leftstr(dl[pos],1)='C' then
begin
    steps:=steps+1;

```

```

        form1.memo2.Lines.Add(inttostr(steps)+' : '+dl[pos]+'消费一个，缓冲减一，队列减一');
    for i:=0 to 14 do
    begin
        if hcdl[i]<>" then
        begin
            form1.memo1.Lines.Add(dl[pos]+'消费： '+hcdl[i]);
            for t:=pos to 13 do
            begin
                hcdl[t]:=hcdl[t+1];           //队列中撤消进程
                hcdl[t+1]:="";
                dl[t]:=dl[t+1];
                dl[t+1]:="";
            end;

            form1.ListBox1.Items.Clear;           //更新列表框
            form1.ListBox2.Items.Clear;
            form1.ListBox3.Items.Clear;
            for t:=0 to 14 do
            begin
                form1.ListBox1.Items.Add(hcdl[t]);
                form1.ListBox2.Items.Add(zsdl[t]);
                form1.ListBox3.Items.Add(jxdl[t]);
            end;
            exit;
        end;
    end;

end;

end;

end;

//// 过程，将指定队列中指定元素送入阻塞队列
procedure ZS(var dl:MyArray;pos:integer);
var
    i,t:integer;
begin
    for i:=0 to 14 do
    begin
        if zsdl[i]=" then
        begin
            zsdl[i]:=dl[pos]; //加入阻塞
            for t:=pos to 13 do
            begin
                dl[t]:=dl[t+1];           //队列中撤消进程
                dl[t+1]:="";
            end;

            form1.ListBox1.Items.Clear;           //更新列表框
            form1.ListBox2.Items.Clear;
            form1.ListBox3.Items.Clear;
            for t:=0 to 14 do
            begin
                form1.ListBox1.Items.Add(hcdl[t]);
                form1.ListBox2.Items.Add(zsdl[t]);
                form1.ListBox3.Items.Add(jxdl[t]);
            end;

            exit;
        end;
    end;

end;

end;

////////////////////////////////////

end.

```

五、运行结果

