Picat: uma linguagem de programação em lógica e multiparadigma

De Sá et al.

UDESC/...

15 de julho de 2017

Sumário

Histórico

- Criada em 2013 por Neng-Fa Zhou e Jonathan Fruhman
- Utiliza B-Prolog como base de implementação, e ambas utilizam a programação em lógica (Lógica de Primeira Ordem)
- Uma evolução do Prolog após mais de 40 anos
- Sua atual versão é a 2.1 (visitado em 15 de julho de 2017)

Ambiente

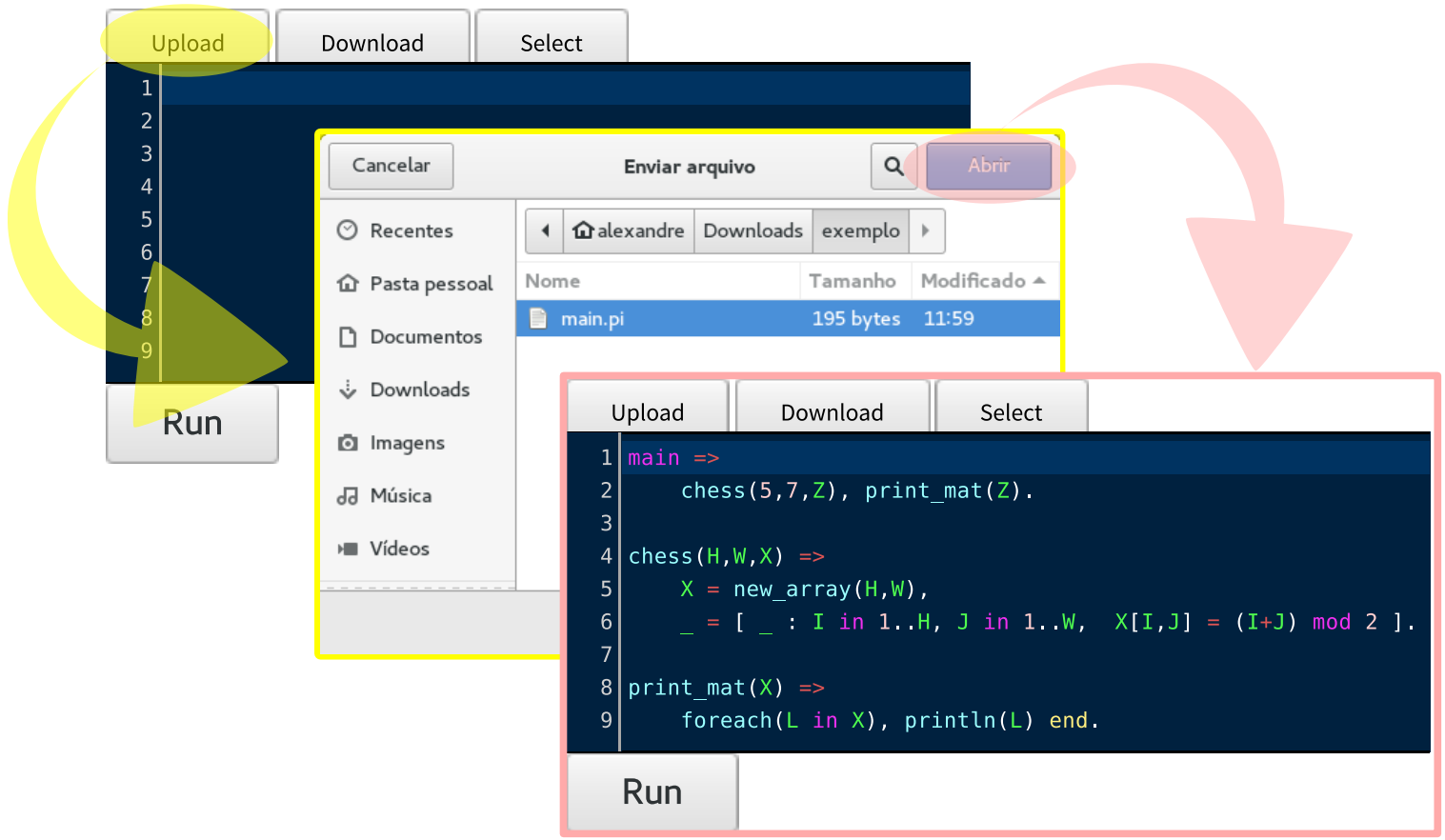
- Criado no primeiro semestre de 2017
  - `http://picat.retina.ufsc.br/picat.html`

Ambiente

- Utiliza CSS/Javascript **CodeMirror** para realce de sintaxe (*syntax highlighting*), endentação e outras formatações
  - `http://codemirror.net/`
- Utiliza **Python** 2.X para uso de CGI (*Common Gateway Interface*), análise preliminar de código-fonte e proteção do servidor (*sandbox*)
  - `https://www.python.org/`
- Utiliza **Picat** 2.1 para execução dos scripts
  - `http://picat-lang.org/`

Upload

- Interface



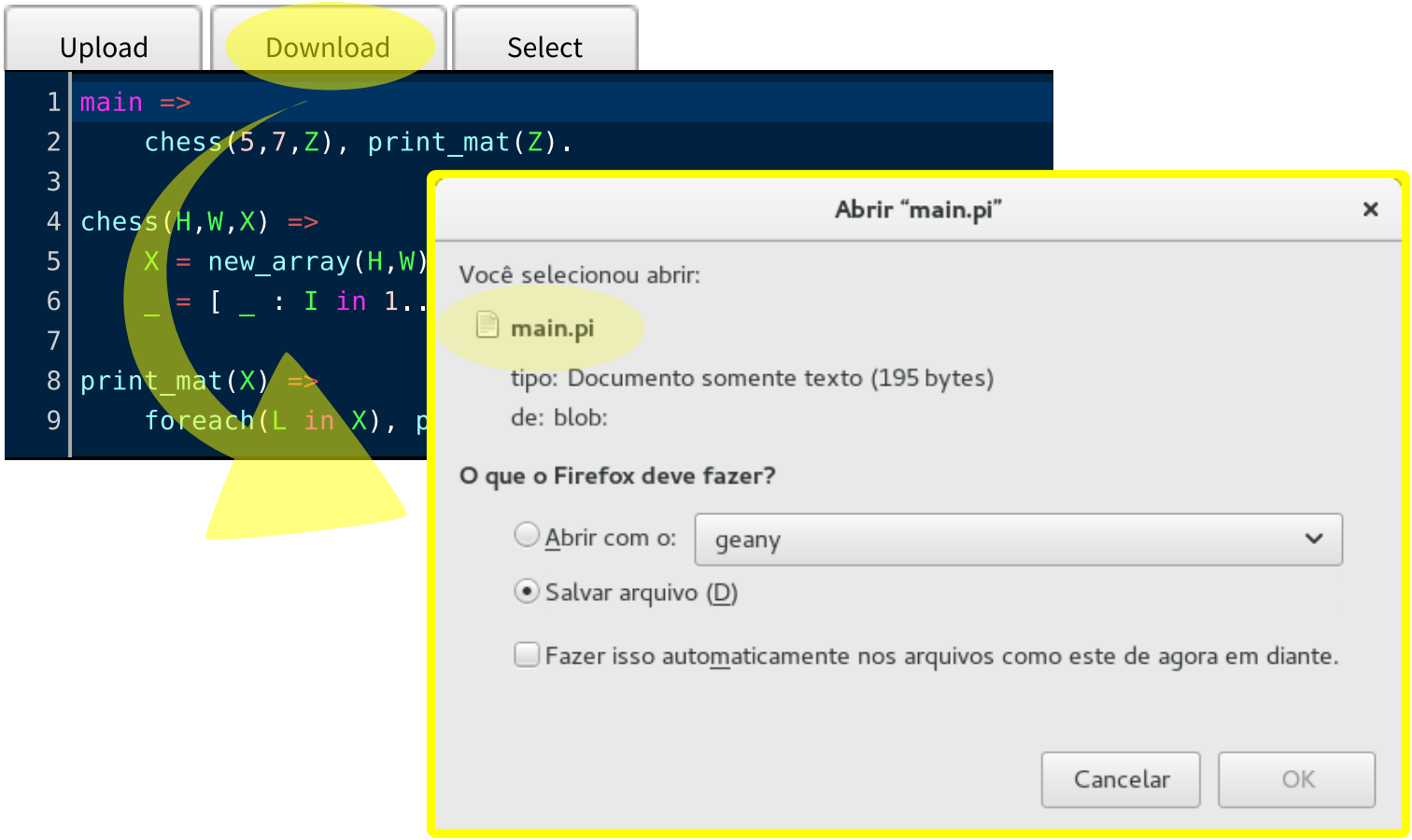
Upload

- Código

```
1  <label for="upload-file">Upload</label><input type="file" onchange="loadfile(this)"
       id="upload-file" style="display:none">
2
3  <script>
4      var editor = CodeMirror.fromTextArea(document.getElementById("code"), {
5          lineNumbers: true,
6          styleActiveLine: true,
7          matchBrackets: true,
8          extraKeys: {"Tab":  "indentAuto"},
9          theme: "erlang-dark"
10     });
11
12     function loadfile(input) {
13         var reader = new FileReader();
14         reader.onload = function(e) {
15             document.getElementById('code').value = e.target.result;
16             editor.setValue(e.target.result);
17         }
18         reader.readAsText(input.files[0]);
19     }
20 </script>
```

Download

- Interface



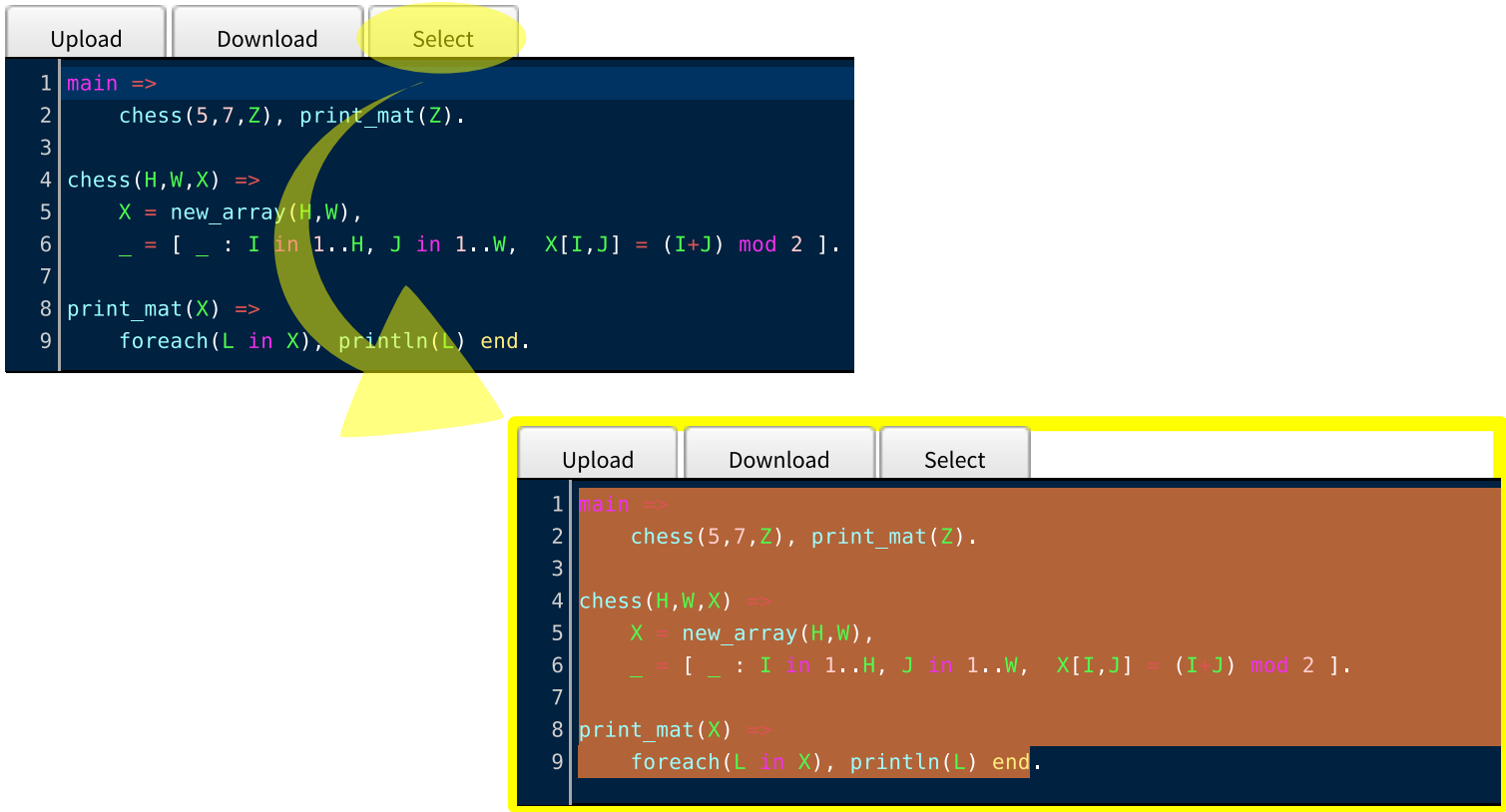
Download

- Código

```
1  <label for="download-file">Download</label><input onclick="saveTextAsFile()" id="
       download-file" style="display:none">
2
3  <script>
4      function saveTextAsFile() {
5          var textToWrite = document.getElementById("code").value;
6          var textFileAsBlob = new Blob([textToWrite], {type:"text/
               plain;charset=utf-8"});
7          var fileNameToSaveAs = "main.pi";
8
9          var downloadLink = document.createElement("a");
10         downloadLink.download = fileNameToSaveAs;
11         downloadLink.innerHTML = "Download File";
12         if (window.webkitURL != null) {
13             downloadLink.href = window.webkitURL.createObjectURL(textFileAsBlob);
14         }
15         else {
16             downloadLink.href = window.URL.createObjectURL(textFileAsBlob);
17             downloadLink.onclick = destroyClickedElement;
18             downloadLink.style.display = "none";
19             document.body.appendChild(downloadLink);
20         }
21
22         downloadLink.click();
23     }
24 </script>
```

Select

- Interface



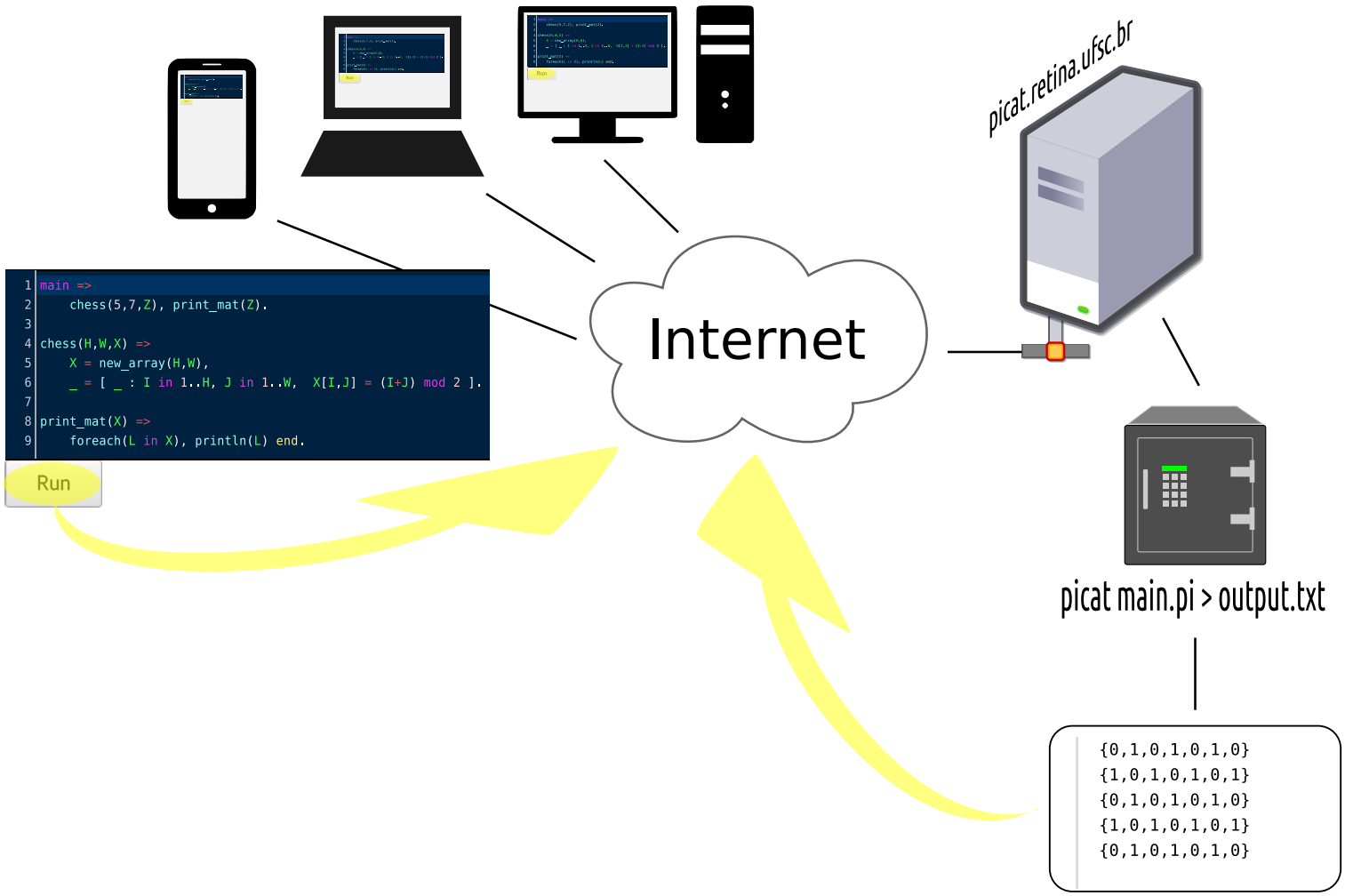
Select

- Código

```
1 <label for="select-file">Select</label><input onclick="mySelectAll()" id="
      select-file" style="display:none">
2
3 <script>
4     function mySelectAll() {
5         editor.execCommand("selectAll");
6         document.execCommand("copy");
7     }
8 </script>
```

Run

- Funcionamento



Run

- Código para execução do script Picat

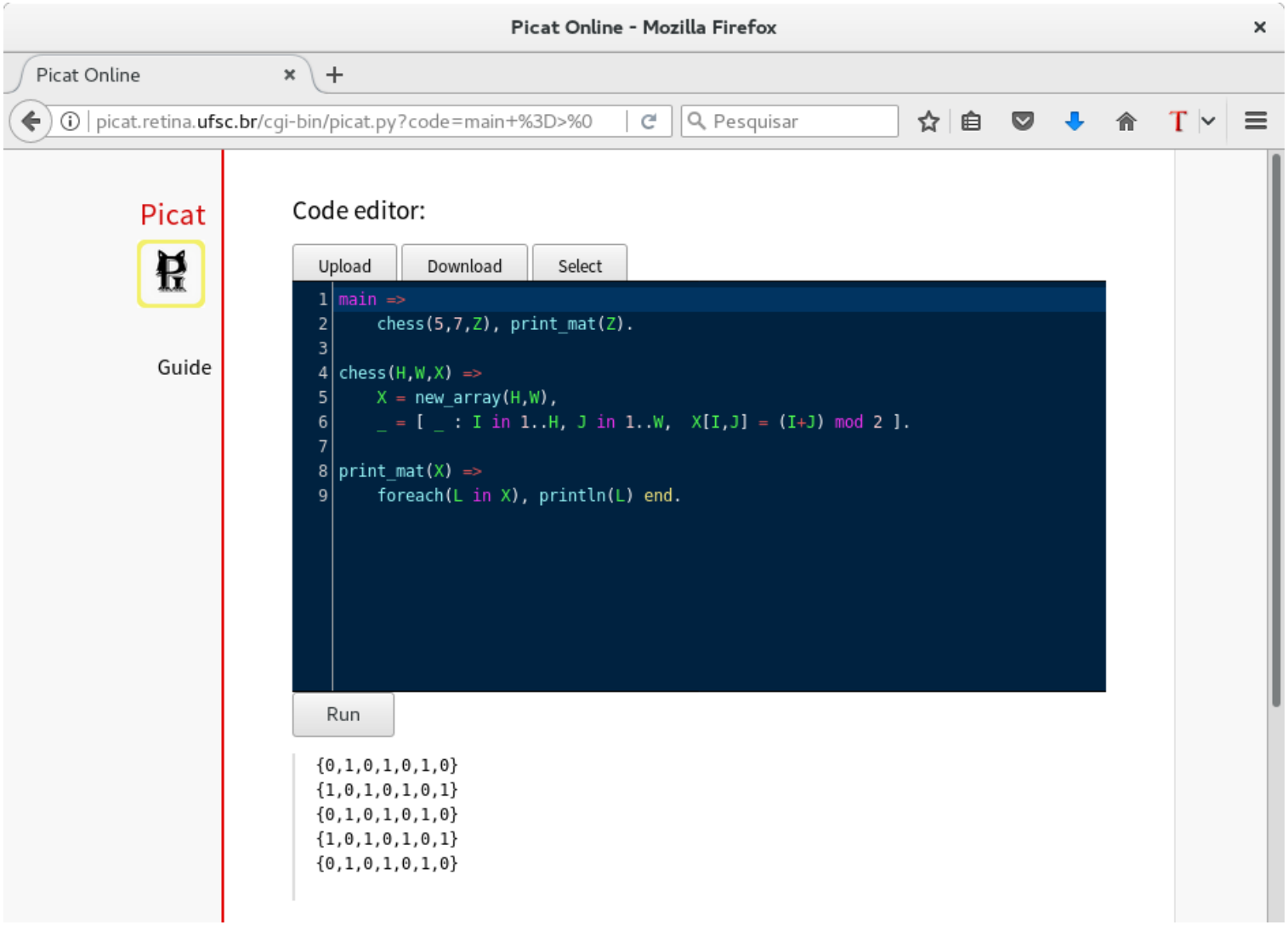
```
1     if re.findall('import\s*os', codepi) or re.findall('command\s*\(', codepi) or
          re.findall('apply\s*\(', codepi):
2         codeout = 'Module "os" and predicates "command/apply" not currently
              available (code not executed)'
3     else:
4         os.system('picat %s > %s' %(CODEPI %(NN), CODEOUT %(NN)))
5         f = open(CODEOUT %(NN), 'r')
6         codeout = f.read()
7         f.close()
8         if ( os.path.getsize(CODEOUT %(NN)) / (1024.0**2) ) > 1:
9             os.remove(CODEOUT %(NN))  #output size over 1 MB is not generated.
```

Run

- Código para criação do HTML (por meio de CGI–Python) com o resultado da execução

```
1     f = open(HTMLBASE, 'r')
2     htm = f.read()
3     f.close()
4
5     htm = htm.split('<!-- split marker -->')
6     htm = '''%s
7 <!-- split marker -->
8 <textarea id="code" name="code">
9 %s</textarea>
10 <!-- split marker -->
11 %s
12 <!-- split marker -->
13 <pre style="width:580px;overflow:auto">
14 %s
15 </pre>
16 <!-- split marker -->
17 %s
18 ''' %(htm[0],codepi,htm[2],codeout,htm[-1])
19
20    print htm
```

Resultado


Picat Online - Mozilla Firefox
Picat Online
picat.retina.ufsc.br/cgi-bin/picat.py?code=main+%3D>%0
Pesquisar
Picat
Guide
Code editor:
Upload
Download
Select
1 main =>
2 chess(5,7,Z), print_mat(Z).
3
4 chess(H,W,X) =>
5 X = new_array(H,W),
6 _ = [_ : I in 1..H, J in 1..W, X[I,J] = (I+J) mod 2].
7
8 print_mat(X) =>
9 foreach(L in X), println(L) end.
Run
{0,1,0,1,0,1,0}
{1,0,1,0,1,0,1}
{0,1,0,1,0,1,0}
{1,0,1,0,1,0,1}
{0,1,0,1,0,1,0}