Charm++ - Bug #1991

CkScanf broken with charmrun

10/10/2018 01:58 PM - Venkatasubrahmanian Narayanan

<table>
<thead>
<tr>
<th>Status:</th>
<th>New</th>
<th>Start date:</th>
<th>10/10/2018</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
<td>Due date:</td>
<td></td>
</tr>
<tr>
<td>Assignee:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Category:</td>
<td></td>
<td>% Done:</td>
<td>0%</td>
</tr>
<tr>
<td>Target version:</td>
<td></td>
<td>Estimated time:</td>
<td>0.00 hour</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Spent time:</td>
<td>0.00 hour</td>
</tr>
</tbody>
</table>

Description

CkScanf fails with an error message "fscanf() failed!" when used in a program run with charmrun. Some debugging suggests that it originates from util/charmrun-src/charmrun.C, from the method input_scanf_chars.

When the code using CkScanf is changed to use scanf or cin, the program appears to not wait for the user to give input (some tests suggest that this might actually be due to the scanf/cin call failing), but just goes ahead with execution.

Reproducible example:

particle.ci:

```cpp
mainmodule particle {
mainchare main {
entry main(CkArgMsg*);
};
};
```

particle.C:

```cpp
#include "particle.decl.h"
class main : public CBase_main {
public:
main(CkArgMsg*) {
int temp;
CkScanf("%d", &temp);
CkPrintf("%d", temp);
CkExit();
}
};
#include "particle.def.h"
```

History

#1 - 10/11/2018 08:59 AM - Sam White

Note that you don't want to use scanf directly because you'll get charmrun and its arguments in the input string. CkScanf is supposed to filter those out.