Wednesday 7/20 Class

- A few brief comments and then I'll turn you loose on Session 20. (Also, questions on Assignment #3.)
  - Spinbox vs. Textedit. Widget. Editor gives hints as questions come in.
- Gameplan for the rest of the quarter. Each of the following is one week:
  - Lipmann-Schwinger equation (brief), path optimization
  - Monte Carlo: Metropolis, spin systems
  - Monte Carlo: Simulated annealing, variational MC
  - C++ classes, C++ classes
  - Parallel processing, with MPI

- Read Chap. 17 handout for Monday

- Quick notes on Fortran calls:
  - In C (as opposed to C), surround prototype of Fortran function with extern "C":
    ```
    extern "C"
    {
    void dgeev_r(
    ```
  - Underscore after name
  - Every argument passed is an address
    ```
    define CHARACTER*1 JOBVL
    char JOBVL,
    ```
    ```
    define INTGER N
    int dimension
    ```
    ```
    define REAL*8 WR(x)
    double * Eigval_real = new double [dimension]
    ```
    ```
    define WR
    double Eigval_real[] Eigval_real,
    ```
Let's try that chart again:

**C++**

<table>
<thead>
<tr>
<th>Defined</th>
<th>Passed</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHARACTER:</td>
<td>JOBVL</td>
</tr>
<tr>
<td>INTEGER N</td>
<td>N</td>
</tr>
<tr>
<td>REAL x3</td>
<td>WR</td>
</tr>
</tbody>
</table>

```cpp
#include <iostream>

int main() {
    int n;
    double wr;
    double jobvl;
    std::cout << "Enter dimension N: " << std::endl;
    std::cin >> n;
    std::cout << "Enter jobvl: " << std::endl;
    std::cin >> jobvl;

    // Do calculations here...

    return 0;
}
```

Dynamically allocating space:

- We want to allocate `F[i]` with space for `maxsize` elements:

```cpp
double* f = new double[maxsize];
```

- Then we can refer to `F[0], F[1], ..., F[maxsize-1]`.

To deallocate `F[i]`:

```cpp
delete [] f;
```

- No need.

For an array `A[i][j]` that is `maxsize` by `maxsize`:

```cpp
double** A = new double*[maxsize];  // Double pointers
```

```cpp
for (int i=0; i < maxsize; i++)
    A[i] = new double[maxsize];
```

- Then we can refer to `A[0][0], A[0][1], A[0][2], and so on`.

To deallocate `A[i][j]`, go backwards:

```cpp
for (int i=0; i < maxsize; i++)
    delete [] A[i];
```

- `delete [] A;`