ICS 01.040.35
CCS L 70

DB35

福 建 省 地 方 标 准

DB35/T 2223—2024

建设工程招投标可验证随机抽取数学模型应用要求

Application requirements of verifiable random selection mathematical model in construction engineering bidding

2024－10－21 发布　　2025－01－21 实施

福建省市场监督管理局　　发 布

目　　次

前　言

本文件按照GB/T 1.1—2020《标准化工作导则　第1部分：标准化文件的结构和起草规则》的规定起草。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别专利的责任。

本文件由厦门市公共资源交易中心提出。

本文件由福建省信息化标准化技术委员会（SAFJ/TC 11）归口。

本文件起草单位：厦门市公共资源交易中心、福建省经济信息中心、福建省公共资源交易公共服务平台有限公司、福建省标准化研究院、福建元素信息科技有限公司、福建首众信息科技有限公司。

本文件主要起草人：林宗昭、许志雄、林平、李志锴、黄志忠、郑楚飞、周顺骥、吴阳、邱培善、陈纯纯、蔡伟文、林荔晶、江峰、张传稀、陈洪。

建设工程招投标可验证随机抽取数学模型应用要求

1 范围

本文件规定了建设工程招投标过程中采用可验证随机抽取数学模型的总体要求、随机抽取流程、模型构建、随机抽取程序要求和验证程序要求。

本文件适用于建设工程招投标过程中采用随机抽取方式确定入围投标人等应用场景。

2 规范性引用文件

本文件没有规范性引用文件。

3 术语和定义

下列术语和定义适用于本文件。

3.1

公共资源交易 public resource trading

工程建设项目、土地使用权、矿业权、国有产权、政府采购、医疗药品及其他公共资源的交易活动。

[来源：DB35/T 2167—2024，3.1]

3.2

公共资源交易电子交易平台 online public resource trading platform

以数据电文形式完成公共资源电子交易活动的信息平台。

注：简称电子交易平台。

[来源：DB35/T 2167—2024，3.3]

3.3

公共资源交易平台运行服务机构 operation service organization of public resources trading

由政府设立或政府通过购买服务等方式确定，通过资源整合共享方式，为公共资源交易相关市场主体、社会公众、行政监督管理部门等提供公共服务的单位。

注：简称“平台运行服务机构”，即公共资源交易中心。

[来源：《公共资源交易平台服务标准（试行）》，3.3]

3.4

随机种子 random seed

在随机数生成过程中使用的、作为随机抽取算法起点的初始值。

3.5

随机抽取程序 random drawing program

一种基于数学模型和算法，通过计算机自动执行随机抽取操作的计算机程序。

3.6

可验证 verifiable

随机抽取数学模型的应用结果能够通过预设的验证程序进行重现和验证。

4 缩略语

下列缩略语适用于本文件。

HASH：散列函数（Hash Function）

JSON：一种轻量级的数据交换格式（Java Script Object Notation）

XML：可扩展标记语言（Extensible Markup Language）

5 总体要求

总体要求如下：

——数学模型及随机抽取程序应融入电子交易平台，实现抽取过程的全流程电子化；

——应提供必要的工具和接口，支持审计机构或监管部门对应用过程进行审计和评估；

——随机抽取过程的关键步骤和参数应完整记录，并可通过验证程序验证随机抽取结果；

——随机抽取过程涉及的重要数据在存储、传输和处理过程应完整和可信。

6 随机抽取流程

随机抽取流程为从确定参与随机抽取的投标人清单到公布结果及验证的全过程，见图1；主要步骤如下：

——招标人或其委托的招标代理机构应根据招标文件的要求确定参与随机抽取的投标人清单和需要随机抽取入围的投标人数量；

——根据“随机种子生成模型”“随机编号生成模型”和“随机抽取结果生成模型”执行随机抽取，生成随机抽取结果并公布；

——招标人或其委托的招标代理机构应通过电子交易平台导出与随机种子组成相关的数据，并当场使用验证程序对随机抽取结果进行验证；

——如果验证结果出现不一致的情况，则应废除之前得出的随机抽取结果，暂停当前交易活动并在问题排查完成后，再次组织重新抽取及验证。

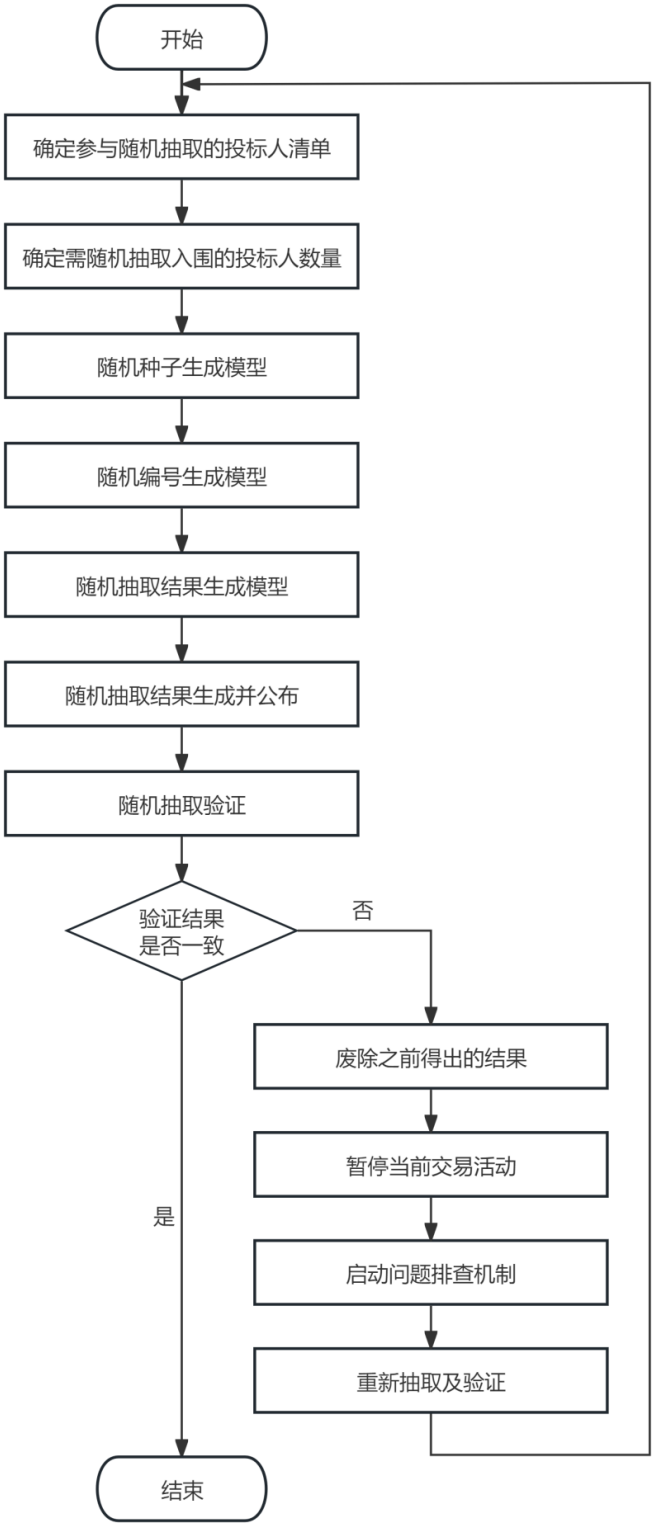
开始
确定参与随机抽取的投标人清单
确定需随机抽取入围的投标人数量
随机种子生成模型
随机编号生成模型
随机抽取结果生成模型
随机抽取结果生成并公布
随机抽取验证
验证结果
是否一致
否
是
废除之前得出的结果
暂停当前交易活动
启动问题排查机制
重新抽取及验证
结束

图1　随机抽取流程图

7 模型构建

7.1 随机种子生成模型

随机种子生成模型以解密成功的电子投标文件受理时间（以服务器上记录时间为准，并将其转换成距离投标当日0时0分0秒的秒数）的平均数四舍五入取整后的整数为基础，在末尾加上参与本次招标的解密成功的投标人数（用4位整数表示），计算生成随机种子。示例见附录A的A.1。

随机种子的生成应具有不可预见性、可验证性。用于随机种子计算的相关参数应在随机抽取时立即公开，并随着招标项目的档案资料进行归档存储，确保项目审计时可从档案材料中获取到相关参数并再次计算出随机种子进行核对。

注：随机种子由开标解密成功的投标人数量及其电子投标文件受理时间两种客观因素计算产生，在开标解密完成前无法确定。随机种子生成的计算参数可替代，但在更改时要重新评估随机种子使用前的不可预见性和使用后的可验证性。更改后的算法要经过严格的测试和验证并公开。

7.2 随机编号生成模型

随机编号生成模型将待抽取的投标人总数假设为n，按某种自然顺序排列后的序号为：1、2、…、n-1、n。在使用随机方式从这n个投标人中产生一定数量入围评审的投标人前，应使用随机编号生成模型对这n个投标人重新排序，为每位投标人分配一个新的随机编号，确保每个抽取对象均可参与抽取过程。模型算法见表1和表2，示例见附录A的A.2。

表1 随机编号生成模型的参数说明

序号	参数	说明
1	a[1)]	模型参数，可使用常数：16807
2	M	模型参数，可使用常数：2^31-1。
3	n	待抽取的投标人总数
4	x_0	正整数，生成投标人随机序号的随机种子
[1)]以上a和M应为互质的正整数，且M应足够大。a和M可替代，但在更改a和M的值时，应重新评估算法的随机性、可预测性。更改后的算法应经过严格的测试和验证并公开。		

表2 随机编号生成模型的算法

步骤	主要内容

计算方法	x ← x_0 # 赋初始值为随机种子
	for k=1,2,...,n
	　　x ← (a*x) mod M # a乘以x，再对M取余数，生成新的随机数
	x ← (a*x) mod M
	$I_{[1]}$ ← floor(x*n/M + 1) # floor为向下取整函数
	k ← 1
	while(k < n)
	{
	　　x ← (a*x) mod M

表 2　随机编号生成模型的算法（续）

步骤	主要内容

计算方法	$I_{[0]}$ ← floor(x*n/M+1)　　# floor为向下取整函数
	# 检查$I_{[0]}$是否已存在于结果数组I中
	if(all($I_{[1:k]}$-$I_{[0]}$!= 0))　　# 如果$I_{[0]}$不在数组I的前k个元素中
	{
	k ← k+1
	$I_{[k]}$← $I_{[0]}$　　# 将 $I_{[0]}$ 添加到结果数组I中
	}
	}

输出结果	$I_1, I_2, \cdots, I_n$　　# 分配给n个投标人的随机序号(按投标人的自然顺序排列)

7.3　随机抽取结果生成模型

在随机编号生成后，n个待抽取的投标人均获得了新的序号。随机结果生成模型将需要抽取入围的投标人数量假设为m（m小于等于n），运用随机方式从n个投标人中抽选m个投标人作为入围投标人抽取结果，保证每个编号在抽取过程中被选中的概率均等。模型算法见表3和表4，示例见附录A的A.3。

表3　随机抽取结果生成模型的参数说明

序号	参数	说明
1	a[2)]	模型参数，可使用常数：69069
2	M	模型参数，可使用常数：2^32-5
3	n	待抽取的投标人总数
4	m	需要抽取入围的投标人数量
5	x_0	正整数，生成入围投标人的随机种子
[2)]以上 a 和 M 应为互质的正整数，且 M 应足够大。a 和 M 可替代，但在更改 a 和 M 的值时，应重新评估算法的随机性、可预测性。更改后的算法应经过严格的测试和验证并公开。		

表4　随机抽取结果生成模型的算法

步骤	主要内容

计算方法	x ← x_0　　# 赋初始值为随机种子
	for　k=1,2,...,n
	x ← (a*x) mod M　　# a乘以x，再对M取余数，生成新的随机数
	x ← (a*x) mod M
	$J_{[1]}$ ← floor(x*n/M + 1)　　# floor为向下取整函数
	k ← 1
	while(k<m)
	{
	x ← (a*x) mod M

表 4　随机抽取结果生成模型的算法（续）

步骤	主要内容

计算方法	$J_{[0]}$ ← floor(x*n/M + 1)　　# floor为向下取整函数
	# 检查$J_{[0]}$是否已存在于结果数组J中
	if (all($J_{[1:k]}-J_{[0]}$!= 0))　# 如果$J_{[0]}$不在数组J的前k个元素中
	{
	k ← k+1
	$J_{[k]}$ ← $J_{[0]}$　　# 将 $J_{[0]}$ 添加到结果数组J中
	}
	}
	J ← sort(J)　　# 对结果数组J进行排序

输出结果	$J_1, J_2, \cdots, J_m$　　# 随机生成的m个入围投标人，按照随机编号生成模型分配的序号从小到大排列

8　随机抽取程序要求

8.1　模型公开要求

满足以下要求：

——所使用的模型及其变更应及时通过平台运行服务机构的门户网站进行公开，包括但不限于随机种子生成模型、随机编号生成模型、随机抽取结果生成模型；

——模型的参数设置、计算方法及解释说明应明确记录在公开文档中。

8.2　应用集成要求

满足以下要求：

——应采用组件化的方式进行开发、维护；

——应提供标准化的接口和文档，并与电子交易平台无缝集成供其调用。

8.3　安全防护要求

满足以下要求：

——在程序运行期间及数据存储等过程应具备防篡改能力；

——应采用国密算法对重要数据进行加密，重要数据包括但不限于随机种子、抽取结果；

——宜使用区块链技术对随机种子等关键参数进行存证。

9　验证程序要求

9.1　公开要求

满足以下要求：

——验证程序应支持离线运行，与验证算法一并通过平台运行服务机构的门户网站公开，供交易主体和社会公众下载和使用；

——在公开验证程序时，应公开该安装包的 HASH 值并加盖平台运行服务机构的电子签章，确保安装包未被篡改；

——验证程序应定期维护，并与模型同步更新。

9.2 数据导入要求

满足以下要求：

——验证程序应支持导入由电子交易平台导出的与随机种子组成相关的数据；

——与随机种子组成相关的数据文件宜采用 XML 文件或 JSON 文件，供交易主体自行验证随机抽取结果。

9.3 验证操作要求

满足以下要求：

——现场验证工作宜由招标人或其委托的招标代理机构负责；

——下载参与随机抽取投标人的电子投标文件受理时间、投标人数量等随机种子组成信息并导入至验证程序；

——执行验证程序并导出随机抽取结果，应与电子交易平台生成的随机抽取结果进行比对，验证是否一致；

——宜提供随机抽取结果生成过程的日志文件供第三方对随机性进行验证，同时使用统计学方法对随机抽取结果进行验证。

验证流程见附录 A 的 A.4，项目应用示例见附录 A 的 A.5。

附 录 A
（资料性）
应用示例

A.1 随机种子生成模型

示例：

```
/**
* Java类
*/
public class Xxx
{
  /**
   * 随机种子的计算
   * @param bidders 参与本次招标的解密成功的投标人数量
   * @return 返回随机种子的字符串
   * @throws ServiceException
   */
  public String generateRandomSeed( List<Record<String, Object>> bidders)
        throws ServiceException
  {
     int n = bidders.size();// 投标人数量
     long j = 0;// 总秒数
     //遍历参与本次招标的解密成功的投标人名单，转换其所有电子投标文件受理时间（以服务器上记录时间为准，并将其转换成距离投标当日0时0分0秒的秒数)为数字并累加总秒数
     for (int i = 0; i < n; i++)
     {
        long s = DateUtils.getTime(bidders.get(i).getString("V_DELIVER_TIME"));
        j += s;
     }
     //计算解密成功的电子投标文件受理时间（以服务器上记录时间为准，并将其转换成距离投标当日0时0分0秒的秒数)的平均数四舍五入取整，并在末尾加上以4位整数表示的参与本次招标的解密成功的投标人数量
     String xx = String.valueOf(Math.round(j * 1.0 / n)) + nextSeqByFour(n);
     logger.info(LogUtils.format("随机种子:" + xx + "种子位数:" + xx.length()));
     return xx ;
  }

  /**
   * 数字前补0至长度为4
   */
  private static String nextSeqByFour(int no)
  {
```

```
        DecimalFormat countFormat = new DecimalFormat("0000");
        return countFormat.format(no);
    }
}
```

A.2 随机编号生成模型

示例：

```
/**
* Java类
*/
public class Xxx
{
  /**
   * 生成投标人随机编号
   * @param x 随机种子
   * @param n 参与本次招标的解密成功的投标人数量
   * @param list 参与本次招标的解密成功的投标人信息列表
   * @return 返回随机编号后的投标人信息列表
   * @throws ServiceException
   */
  public Map<String, Map<String, Object>> assignRandomBidderNumbers(long x, int n,
        List<Map<String, Object>> list) throws ServiceException
  {
    // 处理计算机排序从0开始的问题
    List<Map<String, Object>> newlist = new ArrayList<Map<String, Object>>();
    newlist.add(new HashMap<String, Object>());
    newlist.addAll(list);
    long a = 16807;
    long M = (long) Math.pow(2, 31) - 1;
    for (int i = 1; i <= n; i++)
    {
      x = (a * x) % M;
    }
    x = (a * x) % M;
    int I1 = (int) Math.floor(x * 1.0 / M * n + 1);
    Map<String, Map<String, Object>> resultMap = new LinkedHashMap<String, Map<String, Object>>();
    Map<String, Object> map = newlist.get(1);

    resultMap.put(String.valueOf(I1), map);
    int k = 1;
    while (k < n)
```

```
        {
            x = (a * x) % M;
            int I0 = (int) Math.floor(x * 1.0 / M * n + 1);
            if (!resultMap.containsKey(String.valueOf(I0)))
            {
                k++;
                map = newlist.get(k);
                resultMap.put(String.valueOf(I0), map);
            }
        }
        return resultMap;
    }
}
```

A.3 随机抽取结果生成模型

示例：

```
/**
* Java类
*/
public class Xxx
{
    /**
    * 确定随机抽取入围结果
    * @param x 随机种子
    * @param n 参与本次招标的解密成功的投标人数量
    * @param m 需要抽取入围的投标人数量
    * @param recodeMap 随机编号后的投标人信息列表
    * @return 返回被随机抽中的投标人列表
    * @throws ServiceException
    */
    public Map<String, Map<String, Object>> determineRandomSelectionResults(long x, int n, int m,
            Map<String, Map<String, Object>> recodeMap) throws ServiceException
    {
        long a = 69069;
        long M = (long) Math.pow(2, 32) - 5;
        for (int i = 1; i <= m; i++)
        {
            x = (a * x) % M;
        }
        x = (a * x) % M;
        int J1 = (int) Math.floor(x * 1.0 / M * n + 1);
```

```
        Map<String, Map<String, Object>> resultMap = new LinkedHashMap<String, Map<String, Object>>();
        Map<String, Object> map = recodeMap.get(String.valueOf(J1));
        resultMap.put(String.valueOf(J1), map);
        logger.info(LogUtils.format(map.get("V_BIDDER_NAME").toString()));
        int k = 1;
        while (k < m)
        {
           x = (a * x) % M;
           int IO = (int) Math.floor(x * 1.0 / M * n + 1);
           if (!resultMap.containsKey(String.valueOf(IO)))
           {
              k++;
              map = recodeMap.get(String.valueOf(IO));
              resultMap.put(String.valueOf(IO), map);
              logger.info(LogUtils.format(map.get("V_BIDDER_NAME").toString()));
           }
        }
        return resultMap;
     }
  }
```

A.4 随机抽取结果验证流程

示例：

入围投标人随机抽取的验证流程如下：

a) 下载投递信息

招标人或其委托的招标代理机构在开标辅助系统中导出XML格式的“校验筛选结果的投递信息”文档，该文档包含所有随机种子的组成信息，即参与随机抽取的电子投标文件受理时间、参与随机抽取的投标人数量。

b) 导入数据至离线版

招标人或其委托的招标代理机构打开“离线版”验证程序客户端，导入从开标辅助系统下载的XML格式投标人投递信息文档。

c) 执行随机抽取

招标人或其委托的招标代理机构在“离线版”验证程序客户端中，执行随机抽取，客户端将采用预设的数学模型算法对导入的投标人进行随机抽取。抽取结果将即时显示在客户端界面上，包括入围投标人与未入围投标人的名称。

d) 导出筛选结果

招标人或其委托的招标代理机构在“离线版”验证程序客户端导出Excel格式的筛选结果。

e) 结果比对与验证

招标人或其委托的招标代理机构要同时在开标辅助系统中导出采用数学模型筛选的入围投标人名单Excel表格。将从开标辅助系统下载的入围投标人名单与“离线版”验证程序导出的筛选结果进行比对，验证二者是否一致。可利用Excel的高亮重复项功能进行快速比对。

注：“离线版”验证程序是开源程序，要按本文件“9.3 验证操作要求”提供日志文件生成的代码。

A.5 项目应用示例

示例：

某项目投标人数量为55家，开标解密成功的投标人数量为50家，需要从中抽取20家入围投标人，即取n=50，m=20。参与随机抽取的投标人清单及其电子投标文件受理时间为：

投标人名称	电子投标文件受理时间	投标人名称	电子投标文件受理时间
投标人001	2024-09-24 15:06:44	投标人026	2024-09-26 06:43:27
投标人002	2024-09-24 15:07:43	投标人027	2024-09-26 06:49:28
投标人003	2024-09-24 15:09:31	投标人028	2024-09-26 06:52:39
投标人004	2024-09-24 16:54:56	投标人029	2024-09-26 08:50:36
投标人005	2024-09-24 17:43:26	投标人030	2024-09-26 09:07:00
投标人006	2024-09-24 18:05:25	投标人031	2024-09-26 11:49:39
投标人007	2024-09-25 09:08:14	投标人032	2024-09-26 12:31:21
投标人008	2024-09-25 09:17:25	投标人033	2024-09-26 12:34:56
投标人009	2024-09-25 10:00:41	投标人034	2024-09-26 12:43:38
投标人010	2024-09-25 11:51:38	投标人035	2024-09-26 12:59:20
投标人011	2024-09-25 13:52:54	投标人036	2024-09-26 13:22:05
投标人012	2024-09-25 14:23:00	投标人037	2024-09-26 13:33:28
投标人013	2024-09-25 14:24:10	投标人038	2024-09-26 14:54:27
投标人014	2024-09-25 14:49:00	投标人039	2024-09-26 16:42:09
投标人015	2024-09-25 15:57:05	投标人040	2024-09-26 17:48:21
投标人016	2024-09-26 00:36:48	投标人041	2024-09-26 17:51:13
投标人017	2024-09-26 01:48:58	投标人042	2024-09-26 17:56:56
投标人018	2024-09-26 03:40:04	投标人043	2024-09-26 18:03:19
投标人019	2024-09-26 04:38:12	投标人044	2024-09-26 18:04:08
投标人020	2024-09-26 04:41:42	投标人045	2024-09-26 19:41:46
投标人021	2024-09-26 04:56:35	投标人046	2024-09-26 22:14:39
投标人022	2024-09-26 05:00:19	投标人047	2024-09-26 22:26:55
投标人023	2024-09-26 05:44:08	投标人048	2024-09-26 22:28:52
投标人024	2024-09-26 06:17:39	投标人049	2024-09-26 23:10:49
投标人025	2024-09-26 06:33:59	投标人050	2024-09-26 23:15:23

a) 根据开标解密成功的投标人数量及其电子投标文件受理时间计算生成随机种子，随机种子x0=459680050。

b) 使用参数 a=16807，M=2^31-1，执行随机编号生成模型，生成的随机编号如下：

投标人名称	随机编号	投标人名称	随机编号
投标人 001	41	投标人 026	40
投标人 002	32	投标人 027	31
投标人 003	29	投标人 028	39
投标人 004	5	投标人 029	9

投标人 005	33	投标人 030	43
投标人 006	11	投标人 031	15
投标人 007	48	投标人 032	38
投标人 008	35	投标人 033	24
投标人 009	6	投标人 034	3
投标人 010	36	投标人 035	46
投标人 011	25	投标人 036	10
投标人 012	34	投标人 037	50
投标人 013	18	投标人 038	26
投标人 014	20	投标人 039	37
投标人 015	13	投标人 040	21
投标人 016	30	投标人 041	17
投标人 017	19	投标人 042	45
投标人 018	22	投标人 043	1
投标人 019	2	投标人 044	8
投标人 020	12	投标人 045	23
投标人 021	7	投标人 046	27
投标人 022	49	投标人 047	42
投标人 023	14	投标人 048	16
投标人 024	47	投标人 049	4
投标人 025	28	投标人 050	44

c) 使用参数 a=69069，M=2^32-5，执行随机抽取结果生成模型，生成的随机抽取结果如下：

入围投标人名称	**随机编号**
投标人 034	3
投标人 004	5
投标人 009	6
投标人 021	7
投标人 029	9
投标人 036	10
投标人 040	21
投标人 018	22
投标人 011	25
投标人 038	26
投标人 046	27
投标人 025	28
投标人 016	30
投标人 002	32
投标人 005	33
投标人 010	36

投标人 039	37
投标人 026	40
投标人 030	43
投标人 007	48

d) 验证随机抽取结果，比对随机抽取程序及验证程序生成的入围投标人名单如下，二者一致。

随机抽取程序抽取结果		**验证程序抽取结果**	
入围投标人名称	**随机编号**	**入围投标人名称**	**随机编号**
投标人 034	3	投标人 034	3
投标人 004	5	投标人 004	5
投标人 009	6	投标人 009	6
投标人 021	7	投标人 021	7
投标人 029	9	投标人 029	9
投标人 036	10	投标人 036	10
投标人 040	21	投标人 040	21
投标人 018	22	投标人 018	22
投标人 011	25	投标人 011	25
投标人 038	26	投标人 038	26
投标人 046	27	投标人 046	27
投标人 025	28	投标人 025	28
投标人 016	30	投标人 016	30
投标人 002	32	投标人 002	32
投标人 005	33	投标人 005	33
投标人 010	36	投标人 010	36
投标人 039	37	投标人 039	37
投标人 026	40	投标人 026	40
投标人 030	43	投标人 030	43
投标人 007	48	投标人 007	48

参 考 文 献

[1] DB35/T 2167—2024 政务服务 公共资源交易公共服务系统数据接口要求
[2] 《公共资源交易平台服务标准（试行）》（发改办法规〔2019〕509号）
[3] 《公共资源交易平台管理暂行办法》
[4] 《福建省公共资源交易平台服务标准（2.1）》